

Trabajo de final de grado  
**GRADO DE INGENIERÍA  
INFORMÁTICA**  
Facultad de Matemáticas e Informática  
Universidad de Barcelona

SIDBRINT API/Tool

Autor: Enrique Alonso Miralles Yataco

Director:  
Dr. Simone Balocco  
Responsable: Lourdes Prades Artigas

Barcelona,  
27 de junio de 2019

## Índice General

Abstracto	1
1. Introducción	2
1.1 Motivación	4
1.2 Objetivos	4
1.3 Planificación	5
2. Análisis	9
2.1 Requisitos funcionales	9
2.2 Requisitos no funcionales	10
3. Diseño	10
4. Implementación	29
4.1 Librerías utilizadas a nivel Drupal	29
1. Módulo SERVICES	30
2. Módulo Oauth	33
4.2 Tecnologías usadas a nivel Web	33
1. Laravel	33
2. Bootstrap	36
3. Lealeft.js	38
4. JQuery	38
5. HTML5	39
6. CSS	41
7. Chart.js	42
8. Guzzle	42
9. JavaScript	43
5. Resultados	53
6. Conclusiones	59
7. Trabajo Futuro	59
8. Referencias	60

## Abstracto

Sidbrint es un portal que se encarga de contener y mantener viva la información respecto a la guerra civil española a través de los brigadistas que participaron en esta etapa de la historia española. Al realizar un estudio directo sobre el portal se encontró que las consultas no pueden brindar información directa e interpretativa sin pasar por un proceso de depuración y modelamiento externo; haciendo que la búsqueda de información se convierta en un proceso tedioso.

Como solución a esta problemática se ha fundamentado este proyecto de fin de carrera. Al ver la dificultad contenida en el proceso de la extracción de datos y mantenimiento se estudia la posibilidad de desarrollo de una API. La cual será compartida con los colaboradores a fin de poder ser un nexo para el intercambio de datos de una manera eficaz y práctica para el modelamiento y manejo de resultados a partir de consultas pre condicionadas.

A medida que el proceso sea más eficiente se podrá realizar muchas más variantes en los métodos de la API para poder brindar un uso sólido y seguro para SIDBRINT y sus colaboradores.

## Abstracte

Sidbrint és un portal que s'encarrega d'aportar i mantenir viva la informació sobre la guerra civil espanyola a través dels brigadistes que van participar en aquesta etapa de la història espanyola. Al realitzar un estudi directe sobre el portal es va trobar que les consultes no es podien oferir informació directa i interpretativa sense passar per un procés de depuració i modelació extern fent que la cerca d'informació es converteixi en un procés tediós.

El projecte de final de carrera s'ha funamentat en cercar la solució a aquesta problemàtica. Al veure la dificultat del procés d'obtenció de dades i manteniment s'estudia la possibilitat de desenvolupar una API. Aquesta serà compartida amb els col·laboradors amb la finalitat de ser un enllaç per a l'intercanvi de dades d'una manera eficaç i pràctica pel tractament i utilització dels resultats a partir de consultes precondicionades.

A mesura que el procés sigui més eficient es podrà realitzar moltes més variants en els mètodes de la API per poder oferir un ús sòlid i segur per SIDBRINT i els seus col·laboradors.

## Abstract

Sidbrint is a portal that is responsible for containing and keeping a record on the information regarding the Spanish Civil War through the brigadistas who participated in this stage of the Spanish history. While we were directly working with portal, it was found that the queries can not provide direct and interpretative information without going through a process of external modeling and debugging; making the search for information a tedious process.

As a solution to this problem, this end-of-career project has been defined. Noticing the difficulty to obtain data during the extraction and maintenance, the necessity of developing an API was raised. This New API and the improvement will be shared with the collaborators in order to create a nexus

with them for the exchange of data which would be beneficial to both sides due to the major goal is to find an effective and practical way to redefine and manage the results from a preconditioned consultation.

As the process becomes more efficient, many more variants of API methods can be made in order to provide a solid and secure use of the portal SIDBRINT and its collaborators.

## Introducción

SIDBRINT nació como un proyecto de búsqueda, financiado por el plan nacional de investigación fundamental no orientada del Ministerio de Ciencia e Innovación (HAR2010 – 20983) desarrollado entre el 2010 y 2013, actualmente es un portal institucional de la Universidad de Barcelona.

Lo anterior es una definición obtenida de la misma página, pero para definir correctamente que es SIDBRINT después del desarrollo de este proyecto tendría que mencionar que no es solo un proyecto de búsqueda, es un portal de investigación. Desarrollado colaborativamente por una gran cantidad de investigadores y colaboradores que mantienen día a día la información actualizada con datos sobre las distintas personas que colaboraron en la guerra civil española.

La guerra civil española o guerra de España, también denominada por los españoles como Guerra Civil por antonomasia, fue un conflicto bélico que más tarde repercutiría también en una crisis económica que se desencadenó en España tras el fracaso parcial del golpe de Estado del 17 y 18 de julio de 1936 llevado a cabo por una parte del Ejército contra el Gobierno de la Segunda República. Tras el bloqueo del Estrecho y el posterior puente aéreo que, gracias a la rápida colaboración de la Alemania nazi y la Italia fascista, trasladó las tropas rebeldes a la España peninsular en las últimas semanas de julio, comenzó una guerra civil que concluiría el 1 de abril de 1939 con el último parte de guerra firmado por Francisco Franco, declarando su victoria y estableciendo una dictadura que duraría hasta su muerte el 20 de noviembre de 1975.

La guerra tuvo múltiples facetas, pues incluyó lucha de clases, guerra de religión, enfrentamiento de nacionalismos opuestos, lucha entre dictadura militar y democracia republicana, entre revolución y contrarrevolución, entre fascismo y comunismo.

A las partes del conflicto se las suele denominar bando republicano y bando sublevado:

El bando republicano estuvo constituido en torno al Gobierno, formado por el Frente Popular, que a su vez se componía de una coalición de partidos Republicanos con el Partido Socialista Obrero Español, a la que se habían sumado los marxistas-leninistas del Partido Comunista de España y el POUM, el Partido Sindicalista de origen anarquista y en Cataluña los nacionalistas de izquierda encabezados por Esquerra Republicana de Catalunya. Era apoyado por el movimiento obrero y los sindicatos UGT y CNT, los cuales también perseguían realizar la revolución social. También se había decantado por el bando republicano el Partido Nacionalista Vasco, cuando las Cortes republicanas estaban a punto de aprobar el Estatuto de Autonomía para el País Vasco.

El bando sublevado, que se llamó a sí mismo “bando nacional”, estuvo organizado en torno a parte del alto mando militar, institucionalizado inicialmente en la Junta de Defensa Nacional sustituida tras el nombramiento de Francisco Franco como generalísimo y jefe del Gobierno del Estado.

Políticamente, estuvo integrado por la fascista Falange Española, los carlistas, los monárquicos alfonsinos de Renovación Española y gran parte de los votantes de la CEDA, la Liga Regionalista y otros grupos conservadores. Socialmente fue apoyado por aquellas clases a las que la victoria en las urnas del Frente Popular les hizo sentir que peligraba su posición; por la iglesia católica, acosada por la persecución religiosa desatada por parte de la izquierda nada más estallar el conflicto y por pequeños propietarios temerosos de una “revolución del proletariado”. En las regiones menos industrializadas o primordialmente agrícolas, los sublevados también fueron apoyados por numerosos campesinos y obreros de firmes convicciones religiosas.

Ambos bandos cometieron y se acusaron recíprocamente de la comisión de graves crímenes en el frente y en las retaguardias, como sacas de presos, paseos, desapariciones de personas o tribunales extrajudiciales. La dictadura de Franco investigó y condenó severamente los hechos delictivos cometidos en la zona republicana, llegando incluso a instruir una Causa General, todo ello con escasas garantías procesales. Por su parte, los delitos de los vencedores nunca fueron investigados ni enjuiciados durante el franquismo, a pesar de que algunos historiadores y juristas sostienen que hubo un genocidio en el que, además de subvertir el orden institucional, se habría intentado exterminar a la oposición política.

Las consecuencias de la Guerra Civil han marcado en gran medida la historia posterior de España. Es por ello que la información mantenida en SIDBRINT es relevante. Pero debido a su gran demanda por la información contenida, nace una inquietud. ¿Qué tan colaborativa podría llegar a ser SIDBRINT? De esta pregunta nace este TFG, Sidbrint colaborando con otras entidades mediante una API.

## Motivación

SIDBRINT – Portal colaborativo que mantiene información sobre la guerra civil española.

Sidbrint puede llegar a ser una herramienta colaborativa aún más eficaz si se permite la adición y modelamiento de datos obtenidos de una manera más rápida y abstracta. La información podrá ser modelada indistintamente por las herramientas que los colaboradores crean conveniente para la interpretación de los datos obtenidos por los distintos tipos de consulta que se pudiesen generar para tales fines.

Esta motivación de poder modelar los datos de SIBRINT de maneras distintas es la clave esencial para el desarrollo de SIDBRINT API. Herramienta que actualmente podrá desplegar consultas que representan datos abstractos que para el caso de este TFG se modelarán de manera gráfica bajo el punto de vista estadístico.

## Objetivos:

Los objetivos planteados para el desarrollo de este TFG fueron:

- Compartir y mantener información de manera eficaz.
- Paso intermedio entre el portal actual y nueva aplicación web de Sidbrint
- Visualización de geolocalización de los entes principales (Brigadistas).

Al analizar los objetivos planteados podemos concluir que el objetivo principal de este proyecto es poder compartir información de una manera fiable y segura. La cual permita no solo a colaboradores (empresas, centros de investigación, etc.) acceder a la información relevante sino también modelar y tratar la información desde otro punto de vista. La diferencia de criterios es necesaria para resolver las distintas hipótesis que puedan surgir a partir de los datos obtenidos desde Sidbrint.

Sidbrint dispone de información sobre la guerra civil española y sobre todos los brigadistas que colaboraron en dicha guerra. Toda esta información que puede ser consultada desde el portal. Sin embargo, la visualización de estos datos es presentada de una manera predefinida en la cual se puede obtener la información modelada en una estructura típica de una vista web, esto no permite modelar dicha información para poder obtener estadísticas u otro tipo de visualizaciones que

puedan ayudar a los distintos investigadores a resolver dudas, hipótesis consecuentes con los datos almacenados en SIDBRINT.

Para que Sidbrint sea aún más colaborativa nació la idea de crear una API la cual presenta otras maneras de colaborar con la información relevante para los entes adecuados, los cuales puedan modelar de una manera pertinente los datos de acuerdo a cada tipo de casuística presentada por los distintos colaboradores.

## **Planificación**

Para la planificación del desarrollo del TFG se ha usado el diagrama de GANTT para poder tener gráficamente el detalle donde se estipula las horas invertidas en cada apartado que ha intervenido en el desarrollo total del proyecto.

### **Tareas a realizar:**

#### **Análisis para el desarrollo del TFG**

#### **Estudio e investigación de tecnologías usadas en el desarrollo del proyecto**

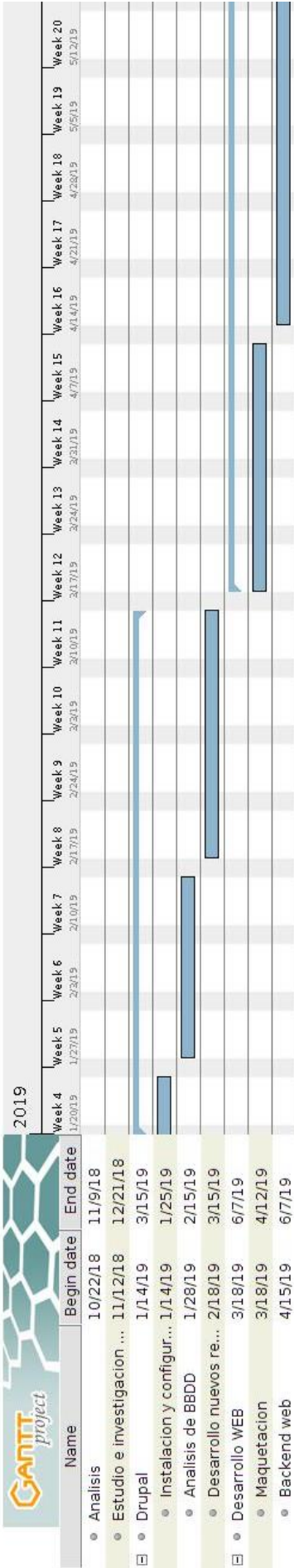
##### **Entorno Laravel API**

- Generación de nuevas relaciones y entidades para la nueva BBDD obtenida a partir de la usada por Drupal en MYSQL.
- Creación de API REST.
  - Análisis y modelamiento de entidades.
  - Seguridad
  - Implementar y estructurar según Restfull cada servicio.
    - Métodos GET, POST, PUT, DELETE.

##### **Entorno Larevel Web**

- Desarrollo de web demostrativa para que interprete los datos de la API y que sea un paso intermedio para la generación de nueva web de Sidbrint.
- Parte Administrativa.
- Parte usuario común.

A continuación, se adjunta el diagrama de planificación original a desarrollar:





Por motivos de mantener la página actual de Sidbrint y la mantención de la estructura actual se decide juntamente con los responsables el de realizar la creación del API usando los elementos que brinda Drupal para tal fin, aunque se deba manejar las limitaciones de estos. A continuación, se detallan las tareas que al final se han realizado.

## **Tareas Realizadas:**

### **Análisis para el desarrollo del TFG**

#### **Estudio e investigación de tecnologías usadas en el desarrollo del proyecto**

##### **Entorno Drupal**

- Instalación y configuración del módulo SERVICES y los submódulos necesarios para que Drupal pueda trabajar mediante webservices y un servidor REST.
- Configuración de las vistas (*views*) necesarias para poder extraer **TODA la información** que existe actualmente en la página web (brigadistas, encuadres militares y fuentes documentales). Estas vistas deberían extraerse en formato JSON.
- Instalación y configuración de un sistema de autenticación segura OAuth, para que las llamadas funcionen con un protocolo de seguridad y a través de una *Auth Key* y una *Secret Key*.

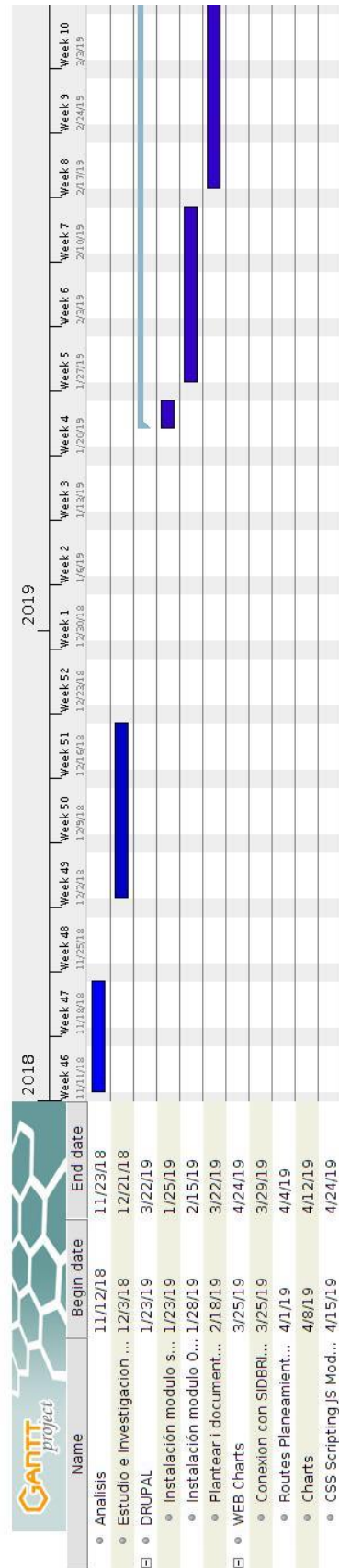
##### **API**

- Plantear y documentar el esquema de llamadas POST/GET de toda la API para acceder a toda la información que contiene SIDBRINT, cada llamada debe estar bien documentada con el código, las opciones que puedan pasarse por parámetros y las posibles respuestas.
- Plantear la API para PHP y JavaScript.

##### **Entorno Visual / Web**

- Crear una página web con toda la documentación y que permita ejecutar las llamadas.
- Crear un ejemplo de uso de estos datos (por ejemplo un mapa con resultados de alguna consulta).

A continuación se muestra en el diagrama de Gantt de las tareas realizadas:



## Análisis

Para el análisis del desarrollo del proyecto se han tenido en cuenta las siguientes necesidades primordiales en el desarrollo.

- Comunicación fluida con los entes colaboradores de SIDBRINT.
- Poder comunicarse con distintos tipos de entes y diversidad de soluciones.
- Simplicidad de mantenimiento.
- Estabilidad en el desarrollo.
- Seguridad en la comunicación.

Para dichos fines se han desarrollado los siguientes requisitos:

### Requisitos funcionales

- LogIn con Drupal (SIDBRINT)

Este requisito hace referencia al proceso que debe seguir el usuario de autenticarse en la aplicación usando su cuenta de SIDBRINT, pudiendo acceder el sistema a sus datos del portal de SIDBRINT.

- LogOut con Drupal (SIDBRINT)

Este requisito hace referencia a la posibilidad de borrar los datos de sesión del usuario en el sistema y así poder entrar con otra cuenta.

- Crear Brigadista

Este requisito hace referencia a realizar una nueva generación del nodo Brigadista sobre el cual se trabaja en SIDBRINT portal.

- Modificar Brigadista

Este requisito hace referencia a la posibilidad de modificar ciertos datos de un nodo brigadista.

- Buscar Brigadistas

Este requisito permite realizar una búsqueda de Brigadistas por ID y o Nombre del brigadista.

- Visualizar Datos Relevantes de Brigadistas

Este requisito permite visualizar datos relevantes respecto a los brigadistas para estadísticas diversas.

- Geolocalización.

Este requisito hace referencia a la Geolocalización de distintos tipos relevantes de características respecto a los brigadistas.

#### Requisitos no funcionales

- Mantenibilidad

El uso de patrones de diseño y de formas óptimas de desarrollar ha permitido mejorar la mantenibilidad del código haciéndolo más rentable y fácil de modificar, además de localizar errores y solucionarlos en el menor tiempo posible.

- Seguridad

La utilización de una librería de autenticación tan conocida como Auth ha permitido desarrollar una aplicación que trata con datos personales de forma segura. Por otro lado se ha añadido un pequeño módulo de seguridad que comprueba la autenticación del usuario y el acceso a los servicios proporcionados. Para ello utiliza el ruteo y la redirección de URLs.

- Escalabilidad

El hecho de haber desarrollado una aplicación mantenible y fácilmente modificable hace que su escalabilidad aumente proporcionalmente. De esta forma es muy sencillo añadir nuevas funciones

## Diseño

Después de ver los requisitos que se tienen que cumplir, en este capítulo se describen las especificaciones de la API. Además, como las APIs se basan en el estilo arquitectónico REST deben cumplir con los principios de diseño de esta arquitectura.

Especificación de la API, existen dos formas de implementar servicios web: SOAP y REST. SOAP tiene la ventaja de ser un estándar y contar con un lenguaje para la descripción de servicios (WSDL), pero en la práctica es más difícil de implementar debido a su gran complejidad. Por ello, seguiremos el estilo arquitectónico REST que es mucho más ligero y fácil de implementar. Además, REST no obliga a utilizar un lenguaje de descripción específico. Para las descripciones utilizaremos el formato JSON ya que es muy sencillo y su uso está muy extendido. Para cumplir con los objetivos fijados en el análisis a continuación se detalla la especificación de la API. En esta especificación se explica qué llamadas se pueden realizar, cómo se pueden realizar y la respuesta que devuelven. Además de esto, el formato de la descripción de un servicio debe estar estructurado como un JSON y contener las claves que se especifican en la siguiente tabla.

Ejemplo de servicio:

Estructura de la descripción de un servicio			
Clave	Tipo	Valor de ejemplo	Descripción
Id	Int	1	Identificador del servicio, se genera automáticamente en el servidor.
Name	String	Nodo	Nombre del Servicio
Method	String	GET	Método HTTP
Descripción	String	Este servicio toma como parámetro un ID para devolver un nodo	La descripción del servicio
EndPoint	String	http://sidbrint.ub.edu/api/rest/nodo/ID	Es una plantilla que hay que completar con los parámetros. Una vez completado es el punto final donde se encuentra el recurso
Parameters	List	Resultado nodo con datos del nodo en sus características principales	Es una lista de parámetros. Cada parámetro consiste en un JSON.

En la siguiente tabla se especifica la estructura de los parámetros del servicio:

Estructura de un parámetro de una descripción		
Clave	Valor de ejemplo	Descripción
Nombre	ID	Valor identificador
Lugar	Path	Lugar donde iría el parámetro
Fecha	01/02/1989	Valor de fecha
Tipo	Entero	Define el tipo de parámetro
Requerido	Sí	Si es un parámetro requerido
Default value	0	Indica el valor por defecto del parámetro
Opciones validas	{1,2,3,...,n}	Indica los valores que son aceptables

Como se ha mencionado anteriormente, en los siguientes puntos se detallan las llamadas que admite la API.

### **Obtener un listado los servicios presentes en la API.**

GET [http://sidbrint.ub.edu/apirest/listado\\_Servicios\\_ApiRest](http://sidbrint.ub.edu/apirest/listado_Servicios_ApiRest)

#### **Parámetros de la llamada:**

Esta llamada no tiene ningún parámetro. Respuestas

#### **Código 200:**

Cuando la petición se ha realizado correctamente se devuelve un listado en formato JSON.

#### **Ejemplo de respuesta:**

```
[
  {
    "nombreServicio": "listado_Brigadistas_PorIds",
    "breveExplicacion": "Para listar uno o varios brigadistas por el identificador único ID o NID",
    "ejemploLLamado":
    "http://sidbrint.ub.edu/apirest/listado_Brigadistas_PorIds?parameters[idstosearch]=a:2:{i:0;i:13300;i:1;i:13307;}"
  },
  {
    "nombreServicio": "geolocalizacion_brigadistas",
    "breveExplicacion": "Localización geográfica por país de nacimiento de los brigadistas",
    "ejemploLLamado": "http://sidbrint.ub.edu/apirest/geolocalizacion_brigadistas"
  },
  {
    "nombreServicio": "numero_Brigadistas_Porgénero_PalabraClave",
    "breveExplicacion": "Cantidad de brigadistas por tipos referentes a palabras claves como ideología política a la que se pertenece",
    "ejemploLLamado":
    "http://sidbrint.ub.edu/apirest/numero_Brigadistas_Porgénero_PalabraClave"
  },
  {
    "nombreServicio":
    "numero_BrigadistasGeolocalizados_PaisNacimiento_PalabraClave",
    "breveExplicacion": "numero de Brigadistas que en el país de procedencia de los distintos brigadistas según palabras clave como ideología política",
    "ejemploLLamado":
    "http://sidbrint.ub.edu/apirest/numero_BrigadistasGeolocalizados_PaisNacimiento_PalabraClave"
  },
  {

```

```

        "nombreServicio": "listado_PalabrasClave",
        "breveExplicacion": "Listado de las palabras clave usadas en Sidbrint para la
        categorización de elementos dentro de la unidad fundamental brigadista",
        "ejemploLLamado": "http://sidbrint.ub.edu/apirest/listado_PalabrasClave"
    }
]

```

**Obtener un listado de los brigadistas totales o por género con su país de nacimiento y las coordenadas para la geolocalización del país.**

GET [http://sidbrint.ub.edu/apirest/geolocalizacion\\_brigadistas](http://sidbrint.ub.edu/apirest/geolocalizacion_brigadistas)

#### **Parámetros de la llamada:**

Esta llamada puede contener parámetros, el parámetro puede ser `parameters[genere]=dona`, `parameters[genere]=home`

#### **Código 200:**

Cuando la petición se ha realizado correctamente se devuelve un listado en formato JSON.

#### **Ejemplo de respuesta:**

```

[
    {
        "Id": 1395,
        "Nombre": "Alemania",
        "Latitud": 51.08342,
        "Longitud": 10.423447,
        "CantidadBrigadistas": 35,
        "arrayTiposBuscados": []
    },
    {
        "Id": 1402,
        "Nombre": "Algèria",
        "Latitud": 28.000027,
        "Longitud": 2.999983,
        "CantidadBrigadistas": 1,
        "arrayTiposBuscados": []
    },
    {
        "Id": 1441,
        "Nombre": "Argentina",
        "Latitud": -34.996496,
        "Longitud": -64.967282,
        "CantidadBrigadistas": 20,
        "arrayTiposBuscados": []
    },
    {

```

```

    "Id": 1466,
    "Nombre": "Austràlia",
    "Latitud": -24.776109,
    "Longitud": 134.755,
    "CantidadBrigadistas": 6,
    "arrayTiposBuscados": []
  },
  {
    "Id": 1420,
    "Nombre": "Àustria",
    "Latitud": 47.200034,
    "Longitud": 13.199959,
    "CantidadBrigadistas": 41,
    "arrayTiposBuscados": []
  },
  {
    "Id": 1404,
    "Nombre": "Bèlgica",
    "Latitud": 50.640281,
    "Longitud": 4.666715,
    "CantidadBrigadistas": 31,
    "arrayTiposBuscados": []
  },
  {
    "Id": 2718,
    "Nombre": "Bolívia",
    "Latitud": -17.05687,
    "Longitud": -64.991229,
    "CantidadBrigadistas": 1,
    "arrayTiposBuscados": []
  },
  {
    "Id": 1439,
    "Nombre": "Brasil",
    "Latitud": -10.333333,
    "Longitud": -53.2,
    "CantidadBrigadistas": 1,
    "arrayTiposBuscados": []
  },
  {
    "Id": 1418,
    "Nombre": "Canadà",
    "Latitud": 61.066692,
    "Longitud": -107.991707,
    "CantidadBrigadistas": 9,
    "arrayTiposBuscados": []
  },
  {
    "Id": 1428,

```



```

    "Nombre": "Cuba",
    "Latitud": 23.013134,
    "Longitud": -80.832875,
    "CantidadBrigadistas": 32,
    "arrayTiposBuscados": []
  },
  {
    "Id": 1396,
    "Nombre": "Espanya",
    "Latitud": 39.326235,
    "Longitud": -4.838065,
    "CantidadBrigadistas": 3,
    "arrayTiposBuscados": []
  },
  {
    "Id": 1392,
    "Nombre": "Estats Units d'Amèrica",
    "Latitud": 39.78373,
    "Longitud": -100.445882,
    "CantidadBrigadistas": 53,
    "arrayTiposBuscados": []
  },
  {
    "Id": 2727,
    "Nombre": "Filipines",
    "Latitud": 12.750349,
    "Longitud": 122.73121,
    "CantidadBrigadistas": 1,
    "arrayTiposBuscados": []
  },
  {
    "Id": 1401,
    "Nombre": "França",
    "Latitud": 46.603354,
    "Longitud": 1.888334,
    "CantidadBrigadistas": 29,
    "arrayTiposBuscados": []
  },
  {
    "Id": 1397,
    "Nombre": "Gran Bretanya",
    "Latitud": 54.702355,
    "Longitud": -3.276575,
    "CantidadBrigadistas": 43,
    "arrayTiposBuscados": []
  },
  {
    "Id": 1415,
    "Nombre": "Hongria",

```

```

    "Latitud": 47.181759,
    "Longitud": 19.506094,
    "CantidadBrigadistas": 1,
    "arrayTiposBuscados": []
  },
  {
    "Id": 1398,
    "Nombre": "Itàlia",
    "Latitud": 41.703519,
    "Longitud": 14.188595,
    "CantidadBrigadistas": 10,
    "arrayTiposBuscados": []
  },
  {
    "Id": 1408,
    "Nombre": "Iugoslàvia",
    "Latitud": 44.817813,
    "Longitud": 20.456897,
    "CantidadBrigadistas": 2,
    "arrayTiposBuscados": []
  },
  {
    "Id": 1431,
    "Nombre": "Lituània",
    "Latitud": 55.35,
    "Longitud": 23.75,
    "CantidadBrigadistas": 1,
    "arrayTiposBuscados": []
  },
  {
    "Id": 1405,
    "Nombre": "Mèxic",
    "Latitud": 19.432601,
    "Longitud": -99.133342,
    "CantidadBrigadistas": 11,
    "arrayTiposBuscados": []
  },
  {
    "Id": 1445,
    "Nombre": "Noruega",
    "Latitud": 64.573154,
    "Longitud": 11.528036,
    "CantidadBrigadistas": 7,
    "arrayTiposBuscados": []
  },
  {
    "Id": 1829,
    "Nombre": "Nova Zelanda",
    "Latitud": -41.500083,

```

```

    "Longitud": 172.834408,
    "CantidadBrigadistas": 3,
    "arrayTiposBuscados": []
  },
  {
    "Id": 1417,
    "Nombre": "Países Baixos",
    "Latitud": 52.237989,
    "Longitud": 5.534607,
    "CantidadBrigadistas": 26,
    "arrayTiposBuscados": []
  },
  {
    "Id": 2767,
    "Nombre": "Palestina",
    "Latitud": 31.649741,
    "Longitud": 35.162072,
    "CantidadBrigadistas": 6,
    "arrayTiposBuscados": []
  },
  {
    "Id": 1403,
    "Nombre": "Polònia",
    "Latitud": 52.097718,
    "Longitud": 19.025816,
    "CantidadBrigadistas": 51,
    "arrayTiposBuscados": []
  },
  {
    "Id": 2398,
    "Nombre": "Puerto Rico",
    "Latitud": 18.164591,
    "Longitud": -66.591798,
    "CantidadBrigadistas": 1,
    "arrayTiposBuscados": []
  },
  {
    "Id": 1394,
    "Nombre": "Romania",
    "Latitud": 45.985213,
    "Longitud": 24.685923,
    "CantidadBrigadistas": 3,
    "arrayTiposBuscados": []
  },
  {
    "Id": 3045,
    "Nombre": "Rússia",
    "Latitud": 64.686314,
    "Longitud": 97.745306,

```

```

    "CantidadBrigadistas": 1,
    "arrayTiposBuscados": []
  },
  {
    "Id": 2745,
    "Nombre": "Sudàfrica",
    "Latitud": -28.816624,
    "Longitud": 24.991639,
    "CantidadBrigadistas": 1,
    "arrayTiposBuscados": []
  },
  {
    "Id": 1419,
    "Nombre": "Suècia",
    "Latitud": 59.674971,
    "Longitud": 14.520858,
    "CantidadBrigadistas": 6,
    "arrayTiposBuscados": []
  },
  {
    "Id": 1425,
    "Nombre": "Suïssa",
    "Latitud": 46.798562,
    "Longitud": 8.231974,
    "CantidadBrigadistas": 33,
    "arrayTiposBuscados": []
  },
  {
    "Id": 1406,
    "Nombre": "Turquia",
    "Latitud": 38.959759,
    "Longitud": 34.924965,
    "CantidadBrigadistas": 1,
    "arrayTiposBuscados": []
  },
  {
    "Id": 1399,
    "Nombre": "Txecoslovàquia",
    "Latitud": 50.087465,
    "Longitud": 14.421254,
    "CantidadBrigadistas": 2,
    "arrayTiposBuscados": []
  },
  {
    "Id": 1400,
    "Nombre": "Unió Soviètica",
    "Latitud": 55.750446,
    "Longitud": 37.617494,
    "CantidadBrigadistas": 24,

```

```

        "arrayTiposBuscados": []
    },
    {
        "Id": 1435,
        "Nombre": "Unió Soviètica (Bielorússia) - Rússia Blanca",
        "Latitud": 53.425061,
        "Longitud": 27.697136,
        "CantidadBrigadistas": 1,
        "arrayTiposBuscados": []
    },
    {
        "Id": 1427,
        "Nombre": "Xile",
        "Latitud": -31.761336,
        "Longitud": -71.31877,
        "CantidadBrigadistas": 1,
        "arrayTiposBuscados": []
    }
]

```

### **Obtener un listado de los brigadistas totales o por género según palabra clave.**

Entre las cuales se pueden usar las siguientes:

Organizaciones y partidos políticos = 2682

Profesiones = 2683

Batallas y Escenarios bélicos = 2651

Campos de concentración y campos de refugiados = 2652

Enquadrament militar de las brigadas internacionales = 2654

Entre otras.

GET

[http://sidbrint.ub.edu/apiarest/numero\\_Brigadistas\\_Porgénero\\_PalabraClave/?parameters\[idPadre\]=2651](http://sidbrint.ub.edu/apiarest/numero_Brigadistas_Porgénero_PalabraClave/?parameters[idPadre]=2651)

GET

[http://sidbrint.ub.edu/apiarest/numero\\_Brigadistas\\_Porgénero\\_PalabraClave/?parameters\[idPadre\]=2651&parameters\[genere\]=dona](http://sidbrint.ub.edu/apiarest/numero_Brigadistas_Porgénero_PalabraClave/?parameters[idPadre]=2651&parameters[genere]=dona)

### **Parámetros de la llamada:**

Esta llamada puede contener parámetros, el parámetro puede ser parameters[genere]=dona, parameters[genere]=home y parameters[idPadre] donde el idPadre es el identificador de palabra clave.

### **Código 200:**

Cuando la petición se ha realizado correctamente se devuelve un listado en formato JSON.

### Ejemplo de respuesta:

[sidbrint.ub.edu/apirest/numero\\_Brigadistas\\_Porgénero\\_PalaraClave/?parameters\[idPadre\]=2651](http://sidbrint.ub.edu/apirest/numero_Brigadistas_Porgénero_PalaraClave/?parameters[idPadre]=2651)

```
[
  {
    "tid": "1603",
    "name": "Batalla de Brunete",
    "count": "447"
  },
  {
    "tid": "1624",
    "name": "Batalla de Belchite",
    "count": "207"
  },
  {
    "tid": "1625",
    "name": "Batalla de Terol",
    "count": "310"
  },
  {
    "tid": "1641",
    "name": "Batalla del Jarama",
    "count": "655"
  },
  {
    "tid": "1646",
    "name": "Batalla de l'Ebre",
    "count": "712"
  },
  {
    "tid": "1652",
    "name": "Batalla de Gadesa",
    "count": "156"
  },
  {
    "tid": "1688",
    "name": "Front de Madrid",
    "count": "293"
  },
  {
    "tid": "1699",
    "name": "Batalla de Llevant",
    "count": "15"
  },
  {
```

```

        "tid": "1701",
        "name": "Batalla de Guadalajara",
        "count": "189"
    },
    etc .....
]

```

[sidbrint.ub.edu/apirest/numero\\_Brigadistas\\_Porgénero\\_PalaraClave/parameters\[idPadre\]=2651&parameters\[genere\]=dona](http://sidbrint.ub.edu/apirest/numero_Brigadistas_Porgénero_PalaraClave/parameters[idPadre]=2651&parameters[genere]=dona)

```

[
    {
        "tid": "1603",
        "name": "Batalla de Brunete",
        "count": "2"
    },
    {
        "tid": "1624",
        "name": "Batalla de Belchite",
        "count": "2"
    },
    {
        "tid": "1625",
        "name": "Batalla de Terol",
        "count": "1"
    },
    {
        "tid": "1641",
        "name": "Batalla del Jarama",
        "count": "3"
    },
    {
        "tid": "1646",
        "name": "Batalla de l'Ebre",
        "count": "2"
    },
    {
        "tid": "1688",
        "name": "Front de Madrid",
        "count": "2"
    },
    {
        "tid": "1701",
        "name": "Batalla de Guadalajara",
        "count": "1"
    },
    {
        "tid": "1706",
        "name": "Front de Còrdova",

```

```

        "count": "2"
    },
    {
        "tid": "1741",
        "name": "Batalla de Madrid",
        "count": "1"
    },
    etc .....
]

```

### **Obtener un listado de las palabras claves usadas en el portal de Sidbrint.**

GET [http://sidbrint.ub.edu/apirest/listado\\_PalabrasClave](http://sidbrint.ub.edu/apirest/listado_PalabrasClave)

#### **Parámetros de la llamada:**

Esta llamada no contiene parámetros.

#### **Código 200:**

Cuando la petición se ha realizado correctamente se devuelve un listado en formato JSON.

#### **Ejemplo de respuesta:**

```

[
    {
        "idtipo": "4",
        "tipo_name": "Enfocament font documental",
        "arrayPadres": [
            "idhijo": "16",
            "hijo_name": "Associacions de brigadistes, col·loquis, congressos i homenatges",
            "array_hijos_hijos": []
        ],
        etc ...
    ],
    {
        "idtipo": "3",
        "tipo_name": "Paraules Clau",
        "arrayPadres": [
            {
                "idparent": "2650",
                "parent_name": "ASSOCIACIONS DE BRIGADISTES, COL·LOQUIS, CONGRESSOS I HOMENATGES",
                "array_hijos": [
                    {
                        "idhijo": "1873",
                        "hijo_name": "Associacions de brigadistes, col·loquis, congressos i homenatges",

```



```

        "array_hijos_hijos": []
    }
]
},
{
    "idparent": "2651",
    "parent_name": "BATALLLES I ESCENARIS BÈL·LICS",
    "array_hijos": [
        {
            "idhijo": "1603",
            "hijo_name": "Batalla de Brunete",
            "array_hijos_hijos": []
        },
        etc .....
    ]
},
{
    "idparent": "2652",
    "parent_name": "CAMPS DE CONCENTRACIÓ, CAMPS DE REFUGIATS I PRESONS",
    "array_hijos": [
        {
            "idhijo": "1630",
            "hijo_name": "Presó militar de les Torres de Cuarte (València)",
            "array_hijos_hijos": []
        },
        etc ....
    ]
},
{
    "idparent": "2654",
    "parent_name": "ENQUADRAMENT MILITAR DE LES BRIGADES INTERNACIONALS",
    "array_hijos": [
        {
            "idhijo": "2657",
            "hijo_name": "- ARTILLERIA",
            "array_hijos_hijos": [
                {
                    "idhijoDeHijo": "2416",
                    "hijo_nameDeHijo": "Base - Maestranza"
                },
                etc .....
            ]
        },
        {
            "idhijo": "2659",
            "hijo_name": "- BATERIA",
            "array_hijos_hijos": [

```

```

{
  "idhijoDeHijo": "2485",
  "hijo_nameDeHijo": "Bateria Agard"
},
etc ....
]
},
{
  "idhijo": "2660",
  "hijo_name": "- BATERIA ANTIAÈRIA_DECA",
  "array_hijos_hijos": [
    {
      "idhijoDeHijo": "2500",
      "hijo_nameDeHijo": "Bateria Antiaèria Alemanya - Gueorgui Dimitrov"
    },
    etc ....,
    {
      "idhijo": "2661",
      "hijo_name": "- BLINDAT",
      "array_hijos_hijos": [
        {
          "idhijoDeHijo": "2506",
          "hijo_nameDeHijo": "Blindat BI XII - Garibaldi"
        },
        etc ...
      ]
    },
    {
      "idhijo": "2664",
      "hijo_name": "- COSSOS DE L'EXÈRCIT",
      "array_hijos_hijos": [
        {
          "idhijoDeHijo": "2594",
          "hijo_nameDeHijo": "Exèrcit de l'aire - Aviació"
        },
        etc ...
      ]
    },
    {
      "idhijo": "2666",
      "hijo_name": "- BATALLÓ",
      "array_hijos_hijos": [
        {
          "idhijoDeHijo": "2437",
          "hijo_nameDeHijo": "Batalló André Marty - Francobelga"
        },
        etc ...
      ]
    },
  ],
}

```

```

    {
      "idhijo": "2668",
      "hijo_name": "- ESCOLA",
      "array_hijos_hijos": [
        {
          "idhijoDeHijo": "2608",
          "hijo_nameDeHijo": "Escuela de Oficiales de Pozorrubio (Albacete)"
        } etc ....
      ]
    },
    {
      "idparent": "2673",
      "parent_name": "ESTATS I TERRITORIS",
      "array_hijos": [
        {
          "idhijo": "1680",
          "hijo_name": "França",
          "array_hijos_hijos": []
        },
        etc ...
      ]
    },
    {
      "idparent": "2674",
      "parent_name": "FORCES ARMADES",
      "array_hijos": [
        {
          "idhijo": "1711",
          "hijo_name": "Légion Tricolore",
          "array_hijos_hijos": []
        },
        etc ....
      ]
    },
    {
      "idparent": "2675",
      "parent_name": "GRUPS HUMANS I PERSONES",
      "array_hijos": [
        {
          "idhijo": "1633",
          "hijo_name": "Jueus",
          "array_hijos_hijos": []
        },
        etc ....
      ]
    },
    {
      "idparent": "2677",
      "parent_name": "HOSPITALS MILITARS I SANITAT DE LES BRIGADES
INTERNACIONALS",
      "array_hijos": [

```

```

    {
      "idhijo": "1684",
      "hijo_name": "Hospital americà",
      "array_hijos_hijos": []
    },
    {
      "idhijo": "1735",
      "hijo_name": "Spanish Medical Aid Committee (SMAC) (Comitè Espanyol d'Ajuda Mèdica)",
      "array_hijos_hijos": []
    },
    {
      "idhijo": "1809",
      "hijo_name": "Hospital de Saragossa",
      "array_hijos_hijos": []
    },
    etc .....
  ]
}
]

```

### **Obtener un listado de la cantidad de brigadistas existentes por palabra clave y género en los países de procedencia.**

Entre las cuales se pueden usar las siguientes:

Organizaciones y partidos políticos = 2682  
 Profesiones = 2683  
 Batallas y Escenarios bélicos = 2651  
 Campos de concentración y campos de refugiados = 2652  
 Enquadrament militar de las brigadas internacionales = 2654  
 Entre otras.

GET

[http://sidbrint.ub.edu/apirest/numero\\_BrigadistasGeolocalizados\\_PaisNacimiento\\_PalabraClave/?parameters\[idPadre\]=2651](http://sidbrint.ub.edu/apirest/numero_BrigadistasGeolocalizados_PaisNacimiento_PalabraClave/?parameters[idPadre]=2651)

GET

[http://sidbrint.ub.edu/apirest/numero\\_BrigadistasGeolocalizados\\_PaisNacimiento\\_PalabraClave/?parameters\[genere\]=dona&parameters\[idPadre\]=2651](http://sidbrint.ub.edu/apirest/numero_BrigadistasGeolocalizados_PaisNacimiento_PalabraClave/?parameters[genere]=dona&parameters[idPadre]=2651)

### **Parámetros de la llamada:**

Esta llamada puede contener parámetros, el parámetro puede ser parameters[genere]=dona, parameters[genere]=home y parameters[idPadre] donde el idPadre es el identificador de palabra clave.

### **Código 200:**

Cuando la petición se ha realizado correctamente se devuelve un listado en formato JSON.

### Ejemplo de respuesta:

[sidbrint.ub.edu/apirest/numero\\_BrigadistasGeolocalizados\\_PaisNacimiento\\_PalabraClave/?parameters\[idPadre\]=2651](http://sidbrint.ub.edu/apirest/numero_BrigadistasGeolocalizados_PaisNacimiento_PalabraClave/?parameters[idPadre]=2651)

```
[
  {
    "Id": 1395,
    "Nombre": "Alemania",
    "Latitud": 51.08342,
    "Longitud": 10.423447,
    "CantidadBrigadistas": 117,
    "arrayTiposBuscados": [
      {
        "Tipo": "Batalla de Brunete",
        "Cantidad": 12
      },
      {
        "Tipo": "Batalla de Belchite",
        "Cantidad": 6
      },
      {
        "Tipo": "Batalla de Terol",
        "Cantidad": 6
      },
      {
        "Tipo": "Batalla del Jarama",
        "Cantidad": 14
      },
      {
        "Tipo": "Batalla de l'Ebre",
        "Cantidad": 23
      },
      etc ....
    ]
  },
  etc ...
]
```

[sidbrint.ub.edu/apirest/numero\\_BrigadistasGeolocalizados\\_PaisNacimiento\\_PalabraClave/?parameters\[genere\]=dona&parameters\[idPadre\]=2651](http://sidbrint.ub.edu/apirest/numero_BrigadistasGeolocalizados_PaisNacimiento_PalabraClave/?parameters[genere]=dona&parameters[idPadre]=2651)

```
[
  {
    "Id": 1395,
```

```

"Nombre": "Alemania",
"Latitud": 51.08342,
"Longitud": 10.423447,
"CantidadBrigadistas": 1,
"arrayTiposBuscados": [
  {
    "Tipo": "Front d'Aragó",
    "Cantidad": 1
  }
],
{
  "Id": 1404,
  "Nombre": "Bèlgica",
  "Latitud": 50.640281,
  "Longitud": 4.666715,
  "CantidadBrigadistas": 1,
  "arrayTiposBuscados": [
    {
      "Tipo": "Front del Nord",
      "Cantidad": 1
    }
  ],
  {
    "Id": 1439,
    "Nombre": "Brasil",
    "Latitud": -10.333333,
    "Longitud": -53.2,
    "CantidadBrigadistas": 1,
    "arrayTiposBuscados": [
      {
        "Tipo": "Batalla d'Oscala",
        "Cantidad": 1
      }
    ],
    {
      "Id": 1428,
      "Nombre": "Cuba",
      "Latitud": 23.013134,
      "Longitud": -80.832875,
      "CantidadBrigadistas": 4,
      "arrayTiposBuscados": [
        {
          "Tipo": "Front de Madrid",
          "Cantidad": 2
        }
      ],
      {

```

```

        "Tipo": "Batalla de Madrid",
        "Cantidad": 1
    },
    {
        "Tipo": "Front d'Aragó",
        "Cantidad": 1
    }
]
},
etc .....
]

```

Nota: Todos los servicios pueden contener más parámetros usando la estructura \$parameters[‘nombre de parámetro’], pero esto conlleva la modificación del código interno de cada servicio para contener la relación de recursos requeridos para dar soporte a la Query realizada.

## Implementación

Después de ver en el capítulo anterior la especificación y el diseño de la API a continuación se presentan los detalles de implementación.

Paquetes utilizados a nivel Drupal:

- Módulo SERVICES.
- Módulo Oauth.

Tecnologías usadas a nivel Web.

- Laravel
- JavaScript
- CSS
- Bootstrap
- Chart.js
- Lealeft.js
- Jquery
- Guzzle

## **Módulo Service**

Es una solución estándar para la creación de APIs que clientes externos pueden usar para la comunicación con Drupal. Su objetivo principal es admitir peticiones a través de HTTP para exponer mediante otros módulos y código características y funcionalidades que contenga Drupal y sus complementos.

La Api del módulo service permite a otros módulos crear otros servicios, incluyendo el control de acceso.

Permite que los módulos creen otros servidores, como SOAP. Se integra con las funcionalidades centrales de Drupal como archivos, nodos, taxonomías, usuarios y más.

El formato de API le permite definir formatos de respuesta para content-type, es decir application/json o application/xml. Los cuales son denominados ENDPOINT.

Los tipos de servidores presentados cuando se agrega el módulo SERVICES son:

REST, XMLRPC, JSONRPC, SOAP, AMF (protocolo RPC flash binario).

### **1 Dependencias de instalación del módulo SERVICE:**

- 1.1 Ctools
- 1.2 Libraries Api
- 1.3 Server REST

### **2 Módulo Ctools :**

Este módulo es principalmente un conjunto de APIs y herramientas para mejorar la experiencia del desarrollador, contiene un módulo administrador de páginas. En particular administra las páginas del panel.

Entre las herramientas que incluye se puede mencionar en particular las siguientes:

- 2.1 Plugins:  
Herramientas para facilitar que los módulos permitan que otros módulos implementen complementos de archivos inc.



- 2.2    Exportables:  
Herramientas para facilitar que los módulos tengan objetos que viven en la base de datos o en el código, como 'vistas predeterminadas'.
- 2.3    AJAX responder:  
Herramientas para facilitar que el servidor maneje las solicitudes de AJAX y le diga al cliente qué hacer con ellas.
- 2.4    Form Tools:  
Herramientas para facilitar el manejo de formularios con AJAX.
- 2.5    Object Caching:  
Herramienta para facilitar la edición de un objeto en varias solicitudes de página y el trabajo de edición en caché.
- 2.6    Contexts:  
La noción de envolver objetos en un contenedor unificado y proporcionar una API para crear y aceptar estos contextos como entrada.
- 2.7    Model Dialog:  
Herramienta para simplificar la colocación de un formulario en un cuadro de diálogo modal.
- 2.8    Dependent:  
Un widget de formulario simple para hacer que los elementos de formulario aparezcan y desaparezcan según las selecciones en otro elemento.
- 2.9    Content:  
Tipos de contenido conectables que se utilizan como paneles en paneles y otros módulos como Dashboard.
- 2.10   Form Wizard:  
Una API para hacer que los formularios de varios pasos sean mucho más fáciles.
- 2.11   CSS Tools:  
Herramientas para almacenar en caché y sanear CSS fácilmente para hacer que el CSS de entrada del usuario sea seguro.

### 3 Libraries API

Este módulo introduce un repositorio común para las bibliotecas. Para los módulos aportados.

sites/all/librariessites/<domain>/libraries

El denominador común para todos los módulos / perfiles / temas de Drupal que se integran con bibliotecas externas.

#### 3.1 Bibliotecas externas.

Indica bibliotecas ("complementos") que no se envían ni se empaquetan con un proyecto en drupal.org. No se quiere alojar bibliotecas de terceros en Drupal por razones de licencias.

Por esta razón el autor del módulo es el creador y el proveedor de la o las bibliotecas que deban utilizarse en los desarrollos de bibliotecas externas y distribuidas.

#### 3.2 Metas definidas por el módulo de librerías API.

- La misma biblioteca puede ser compartida por más de un módulo.
- Se debe facilitar el proceso de actualización de un módulo que requiere una biblioteca externa. Simplemente se ha de reemplazar la carpeta del módulo con la nueva. No es necesario eliminar la biblioteca y volver a registrarla.
- Se ha de prevenir las incompatibilidades relacionadas con la instalación de la biblioteca más de una vez en diferentes versiones.
- Instrucciones de instalación central (widget) para ayudar a los usuarios a descubrir cómo instalar.

#### 3.3 Clasificación de la biblioteca.

#### 3.4 Manejo de dependencias.

#### 3.5 Detección de la versión de la biblioteca.

#### 3.6 Control de tiempo de ejecución de la disponibilidad de la biblioteca.

## 4 Server REST

El servidor REST proporciona una interfaz para que el módulo de Servicios se comuniquen con los clientes REST.

Una vez instalados las dependencias y descargado el módulo SERVICES para su instalación se requiera la activación de los módulos cargados, Server Rest, Ctools, Services.

### Módulo OAuth

**Open Authorization (OAuth)** es un estándar abierto que permite flujos simples de autorización para sitios web o aplicaciones informáticas. Se trata de un protocolo propuesto por Blaine Cook y Chris Messina, que permite autorización segura de una API de modo estándar y simple para aplicaciones de escritorio, móviles y web.

- OAuth permite a un usuario del sitio A compartir su información en el sitio A (proveedor de servicio) con el sitio B (llamado consumidor) sin compartir toda su identidad. Para desarrolladores de consumidores, OAuth es un método de interactuar con datos protegidos y publicarlos. Para desarrolladores de proveedores de servicio, OAuth proporciona a los usuarios un acceso a sus datos al mismo tiempo que protege las credenciales de su cuenta. Este mecanismo es utilizado por compañías como Google, Facebook, Microsoft, Twitter y Github para permitir a los usuarios compartir información sobre sus cuentas con aplicaciones de terceros o sitios web.
- Para obtener este protocolo en Drupal versión 7 se ha de instalar el módulo OAuth 1.0

### Laravel

Laravel es un framework de código abierto para desarrollar aplicaciones y servicios web con PHP 5 y PHP 7. Su filosofía es desarrollar código PHP de forma elegante y simple, evitando el "código espagueti".

Fue creado en 2011 y tiene una gran influencia de frameworks como Ruby on Rails, Sinatra y ASP.NET MVC.

Fundamentado en el patrón de desarrollo MVC.

## 1. MVC

Es un patrón de diseño de software para programación que propone separar el código de los programas por sus diferentes responsabilidades.

En líneas generales, MVC es una propuesta de diseño de software utilizada para implementar sistemas donde se requiere el uso de interfaces de usuario.

Surge de la necesidad de crear software más robusto con un ciclo de vida más adecuado, donde se potencie la facilidad de mantenimiento, reutilización del código y la separación de conceptos.

Su fundamento es la separación del código en tres capas diferentes, acotadas por su responsabilidad, en lo que se llaman Modelos, Vistas y Controladores, o lo que es lo mismo, Model, Views & Controllers, en inglés.

MVC es un "invento" que ya tiene varias décadas y fue presentado incluso antes de la aparición de la Web. No obstante, en los últimos años ha ganado mucha fuerza y seguidores gracias a la aparición de numerosos frameworks de desarrollo web que utilizan el patrón MVC como modelo para la arquitectura de las aplicaciones web.

### ○ Modelos

Es la capa donde se trabaja con los datos, por tanto, contendrá mecanismos para acceder a la información y también para actualizar su estado. Los datos los tendremos habitualmente en una base de datos, por lo que en los modelos tendremos todas las funciones que accederán a las tablas y harán los correspondientes selects, updates, inserts, etc.

No obstante, cabe mencionar que cuando se trabaja con MVC lo habitual también es utilizar otras librerías como PDO o algún ORM como Doctrine, que nos permiten trabajar con abstracción de bases de datos y persistencia en objetos. Por ello, en vez de usar directamente sentencias SQL, que suelen depender del motor de base de datos con el que se esté trabajando, se utiliza un dialecto de acceso a datos basado en clases y objetos.

### ○ Vistas

Las vistas, como su nombre nos da a entender, contienen el código de nuestra aplicación que va a producir la visualización de las interfaces de usuario, o sea, el código que nos permitirá renderizar los estados de nuestra aplicación en HTML. En las vistas nada más tenemos los códigos HTML y PHP que nos permite mostrar la salida.

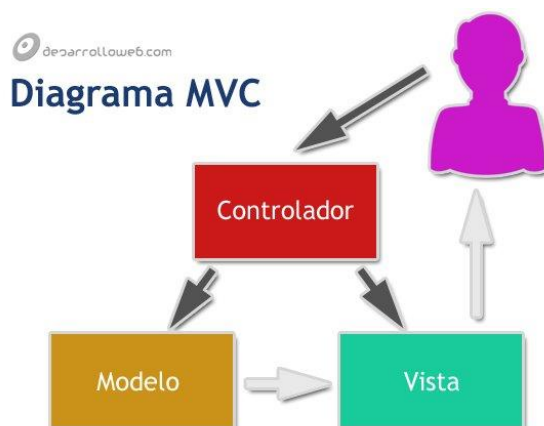
En la vista generalmente trabajamos con los datos, sin embargo, no se realiza un acceso directo a éstos. Las vistas requerirán los datos a los modelos y ellas se generará la salida, tal como nuestra aplicación requiera.

### ○ Controladores

Contiene el código necesario para responder a las acciones que se solicitan en la aplicación, como visualizar un elemento, realizar una compra, una búsqueda de información, etc.

En realidad, es una capa que sirve de enlace entre las vistas y los modelos, respondiendo a los mecanismos que puedan requerirse para implementar las necesidades de nuestra aplicación. Sin embargo, su responsabilidad no es manipular directamente datos, ni mostrar ningún tipo de salida, sino servir de enlace entre los modelos y las vistas para implementar las diversas necesidades del desarrollo.

### Arquitectura de aplicaciones MVC



En la imagen presentada se representa con flechas los modos de colaboración entre los distintos elementos que formarían una aplicación MVC, junto con el usuario. Como se puede ver, los controladores, con su lógica de negocio, hacen de puente entre los modelos y las vistas. Pero además en algunos casos los modelos pueden enviar datos a las vistas. Veamos paso a paso cómo sería el flujo de trabajo característico en un esquema MVC.

- El usuario realiza una solicitud a nuestro sitio web. Generalmente estará desencadenada por acceder a una página de nuestro sitio. Esa solicitud le llega al controlador.
- El controlador comunica tanto con modelos como con vistas. A los modelos les solicita datos o les manda realizar actualizaciones de los datos. A las vistas les solicita la salida correspondiente, una vez se

hayan realizado las operaciones pertinentes según la lógica del negocio.

- Para producir la salida, en ocasiones las vistas pueden solicitar más información a los modelos. En ocasiones, el controlador será el responsable de solicitar todos los datos a los modelos y de enviarlos a las vistas, haciendo de puente entre unos y otros. Sería corriente tanto una cosa como la otra, todo depende de nuestra implementación.
- Las vistas envían al usuario la salida. Aunque en ocasiones esa salida puede ir de vuelta al controlador y sería éste el que hace el envío al cliente.

Laravel como hemos visto es una tecnología para el desarrollo de soluciones web usando un framework bajo el fundamento del protocolo mvc, gracias a esta tecnología se ha desarrollado la parte web donde se presentan las visualizaciones de cada una de las estructuras Json resueltas por los servicios a las peticiones get que se aplican al endpoint que mantiene la Api Sidbrint.

Por ello se usa dentro de la estructura las rutas las cuales permiten realizar las llamadas a los modelos que contienen la funcionalidad para realizar las request a los servicios desplegados mediante la Api Sidbrint realizada con el módulo service en Drupal.

## **Bootstrap**

Es una biblioteca multiplataforma o conjunto de herramientas de código abierto para diseño de sitios y aplicaciones web.

Contiene plantillas de diseño con tipografía, formularios, botones, cuadros, menús de navegación y otros elementos de diseño basado en HTML y CSS para tipografía, formularios, botones, tablas, navegación, modales, carruseles de imágenes y muchos otros, así como extensiones de JavaScript adicionales. A diferencia de muchos frameworks web, solo se ocupa del desarrollo front-end.

Es un marco front-end gratuito para un desarrollo web más rápido y sencillo.

Es el segundo proyecto más destacado en GitHub y es usado por la NASA y la MSNBC entre otras organizaciones.

Bootstrap también te da la posibilidad de crear fácilmente diseños responsive.

Responsive:

El diseño web responsive o adaptativo es una técnica de diseño web que busca la correcta visualización de una misma página en distintos dispositivos. Desde ordenadores de escritorio a tablets y móviles.

Hoy en día accedemos a sitios web desde todo tipo de dispositivos; ordenador, tablet, smartphone... por lo que, cada vez más, nos surge la necesidad de que nuestra web se adapte a los distintos tamaños de pantalla de los dispositivos que usamos día a día.

¿En qué consiste el diseño responsive?

Se trata de redimensionar y colocar los elementos de la web de forma que se adapten al ancho de cada dispositivo permitiendo una correcta visualización y una mejor experiencia de usuario. Se caracteriza porque los layouts (contenidos) e imágenes son fluidos y se usa código media-queries de CSS3.

El diseño responsive permite reducir el tiempo de desarrollo, evita los contenidos duplicados, y aumenta la viralidad de los contenidos ya que permite compartirlos de una forma mucho más rápida y natural.



Se basa en proporcionar a todos los usuarios de una web los mismos contenidos y una experiencia de usuario lo más similar posible, frente a otras aproximaciones al desarrollo web móvil como la creación de apps, el cambio de dominio o webs servidas dinámicamente en función del dispositivo.

El diseño web responsive se consolida como una de las mejores prácticas hoy en día en diseño web. Aunque como todas, tiene sus pros y contras, la web responsive es considerada hoy en día la mejor práctica posible en el diseño web.

Para el TFG se ha realizado usando esta tecnología para que las distintas paginas creadas mediante las clases otorgadas por esta tecnología brinden la característica responsive a todas las vistas.

## **Lealeft**

Es una biblioteca de código fuente abierto de Java Script que se utiliza para crear aplicaciones de mapas web. Lanzado por primera vez en 2011, es compatible con la mayoría de las plataformas móviles y de escritorio, y admite HTML5 y CSS3. Junto con OpenLayers y la API de Google Maps, es una de las bibliotecas de mapas de JavaScript más populares y es utilizada por los principales sitios web como FourSquare, Pinterest y Flickr.

Lealeft permite a los desarrolladores sin un fondo GIS visualizar fácilmente mapas web en mosaico alojados en un servidor público, con superposiciones en mosaico opcionales. Puede cargar datos de entidades de los archivos GeoJSON, personalizarlos y crear capas interactivas, como marcadores con ventanas emergentes cuando se hace clic.

Fue desarrollado por Vladimir Agafonkin, quien se unió a Mapbox en 2013 y ahora este proyecto sigue siendo desarrollado por una gran cantidad de contribuyente pertenecientes a la comunidad de Lealeft.

Lealeft una tecnología usada para gráficos donde se pueden mantener geolocalizaciones a partir de proporcionar valores como latitud y longitud de cada origen, es por ello que se ha utilizado para todas las vistas que contienen geolocalización para usar la estructura proveniente en Json que mantiene la característica necesaria para que pueda ser graficada bajo planos de ubicación.

## **Jquery**

JQuery es una biblioteca multiplataforma de JavaScript, creada inicialmente por John Resig, que permite simplificar la manera de interactuar con los documentos HTML, manipular el árbol DOM, manejar eventos, desarrollar animaciones y agregar interacción con la técnica AJAX a páginas\_web.

Fue presentada el 14 de enero de 2006 en el BarCamp NYC. De acuerdo a un análisis de la Web (realizado en 2017) JQuery es la biblioteca de JavaScript más utilizada, por un amplio margen.

jQuery es software libre y de código abierto, posee un doble licenciamiento bajo la Licencia\_MIT y la Licencia\_Pública General de GNU v2, permitiendo su uso en proyectos libres y privados. jQuery, al igual que otras bibliotecas, ofrece una serie de funcionalidades basadas en JavaScript que de otra manera requerirían de mucho más código, es decir, con las funciones propias de esta biblioteca se logran grandes resultados en menos tiempo y espacio.

La sintaxis de JQuery está diseñada para facilitar la navegación por un documento, seleccionar elementos DOM, crear animaciones, manejar eventos y desarrollar aplicaciones AJAX. JQuery también proporciona capacidades para que los desarrolladores creen complementos en la parte superior de la biblioteca de JavaScript. Esto permite a los desarrolladores crear abstracciones para



interacción y animación de bajo nivel, efectos avanzados y widgets temáticos de alto nivel. El enfoque modular de la biblioteca jQuery permite la creación de poderosas páginas web dinámicas y aplicaciones web.

Gracias a la particularidad que brinda JQuery respecto a las llamadas Ajax sobre URLs, se ha aprovechado esta característica para la petición de datos en su formato Json para la carga dinámica de resultados para las gráficas presentadas en este TFG.

## HTML5

HTML5 (HyperText Markup Language, versión 5) es la quinta revisión importante del lenguaje básico de la World Wide Web, HTML. HTML5 especifica dos variantes de sintaxis para HTML: una «clásica», HTML (text/html), conocida como HTML5, y una variante XHTML conocida como sintaxis XHTML5 que deberá servirse con sintaxis XML (application/xhtml+xml). Esta es la primera vez que HTML y XHTML se han desarrollado en paralelo. La versión definitiva de la quinta revisión del estándar se publicó en octubre de 2014.

Al no ser reconocido en viejas versiones de navegadores por sus nuevas etiquetas, se recomienda al usuario común actualizar su navegador a la versión más nueva, para poder disfrutar de todo el potencial que provee HTML5.

El desarrollo de este lenguaje de marcado es regulado por el Consorcio W3C.

HTML5 establece una serie de nuevos elementos y atributos que reflejan el uso típico de los sitios web modernos. Algunos de ellos son técnicamente similares a las etiquetas <div> y <span>, pero tienen un significado semántico, como por ejemplo <nav> (bloque de navegación del sitio web) y <footer> (Pie de página).

- Novedades:
  - .1. Incorpora etiquetas (canvas 2D y 3D, audio, vídeo) con codecs para mostrar los contenidos multimedia. Actualmente hay una lucha entre imponer codecs libres (WebM + VP8) o privados (H.264/MPEG-4 AVC).
  - .2. Etiquetas para manejar grandes conjuntos de datos: Datagrid, Details, Menu y Command. Permiten generar tablas dinámicas que pueden filtrar, ordenar y ocultar contenido en cliente.

- .3. Mejoras en los formularios. Nuevos tipos de datos (eMail, number, url, datetime ...) y facilidades para validar el contenido sin Javascript.
  - .4. Visores: MathML (fórmulas matemáticas) y SVG (gráficos vectoriales). En general se deja abierto a poder interpretar otros lenguajes XML.
  - .5. Drag & Drop. Nueva funcionalidad para arrastrar objetos como imágenes.
- Web Semántica:
    - .1. Añade etiquetas para manejar la Web semántica (Web 3.0): header, footer, article, nav, time (fecha del contenido), link rel="" (tipo de contenido que se enlaza).
    - .2. Estas etiquetas permiten describir cuál es el significado del contenido. Por ejemplo su importancia, su finalidad y las relaciones que existen. No tienen especial impacto en la visualización, se orientan a buscadores.
    - .3. Los buscadores podrán indexar e interpretar esta metainformación para no buscar simplemente apariciones de palabras en el texto de la página.
    - .4. Permite incorporar a las páginas ficheros RDF / OWL (con metainformación) para describir relaciones entre los términos utilizados.
    - .5. Además, ofrece versatilidad en el manejo y animación de objetos simples, imágenes etc.
  - Nuevas APIs y Javascript:
    - .1. API para hacer Drag & Drop. Mediante eventos.
    - .2. API para trabajar Off-Line. Permite descargar todos los contenidos necesarios y trabajar en local.
    - .3. API de Geolocalización para dispositivos que lo soporten.
    - .4. API Storage. Facilidad de almacenamiento persistente en local, con bases de datos (basadas en SQLite) o con almacenamiento de objetos por aplicación o por dominio Web (Local Storage y Global Storage). Se dispone de una Base de datos con la posibilidad de hacer consultas SQL.

- .5. WebSockets. API de comunicación bidireccional entre páginas. Similar a los Sockets de C.
- .6. WebWorkers. Hilos de ejecución en paralelo.
- Estándar futuro. System Information API. Acceso al hardware a bajo nivel: red, ficheros, CPU, memoria, puertos USB, cámaras, micrófonos... Muy interesante, pero con numerosas salvedades de seguridad.

HTML5 contiene etiquetas que son usadas por distintas librerías JavaScript para que interactúen con las distintas funcionalidades alojadas en cada una de estas librerías. Por ello se ha usado tags de HTML5 para poder usar determinadas características respecto a las librerías usadas en este TFG como son: Lealeft, ChartsJS.

## CSS

CSS (siglas en inglés de Cascading Style Sheets), en español "Hojas de estilo en cascada", es un lenguaje de diseño gráfico para definir y crear la presentación de un documento estructurado escrito en un lenguaje de marcado. Es muy usado para establecer el diseño visual de los documentos web, e interfaces de usuario escritas en HTML o XHTML; el lenguaje puede ser aplicado a cualquier documento XML, incluyendo XHTML, SVG, XUL, RSS, etc. También permite aplicar estilos no visuales, como las hojas de estilo auditivas.

Junto con HTML y JavaScript, CSS es una tecnología usada por muchos sitios web para crear páginas visualmente atractivas, interfaces de usuario para aplicaciones web y GUIs para muchas aplicaciones móviles (como Firefox OS).

CSS está diseñado principalmente para marcar la separación del contenido del documento y la forma de presentación de este, características tales como las capas o layouts, los colores y las fuentes.

Esta separación busca mejorar la accesibilidad del documento, proveer más flexibilidad y control en la especificación de características de presentación, permitir que varios documentos HTML compartan un mismo estilo usando una sola hoja de estilos separada en un archivo .css, y reducir la complejidad y la repetición de código en la estructura del documento.

La separación del formato y el contenido hace posible presentar el mismo documento marcado en diferentes estilos para diferentes métodos de renderizado, como en pantalla, en impresión, en voz (mediante un navegador de voz o un lector de pantalla, y dispositivos táctiles basados en el sistema Braille. También se puede mostrar una página web de manera diferente dependiendo del tamaño de la pantalla o tipo de dispositivo. Los lectores pueden especificar una hoja de estilos diferente, como

una hoja de estilos CSS guardado en su computadora, para sobrescribir la hoja de estilos del diseñador.

La especificación CSS describe un esquema prioritario para determinar qué reglas de estilo se aplican si más de una regla coincide para un elemento en particular. Estas reglas son aplicadas con un sistema llamado de cascada, de modo que las prioridades son calculadas y asignadas a las reglas, así que los resultados son predecibles.

La especificación CSS es mantenida por el World Wide Web Consortium (W3C). El MIME type text/css está registrado para su uso por CSS descrito en el RFC 23185. El W3C proporciona una herramienta de validación de CSS gratuita para los documentos CSS.

CSS nos permite realizar modificaciones respecto a atributos para una mejora en la visualización de los distintos elementos en el DOM de las páginas web. Por esta razón, aunque bootstrap es una librería basada en CSS y Javascript para realizar paginas responsive. Quizás en algunas características no pueda ser usado en su totalidad o no permite por reglas de esta librería que se puedan manejar ciertos atributos, permitiendo así el uso de CSS para modificar lo que por bootstrap no pueda ser manejado y deba ser modificado para brindar una experiencia distinta a la predefinida por la librería de bootstrap. A si en este TFG se ha usado para manipular ciertas características del Layaout de la aplicación web que se ha usado para la presentación de los datos extraídos desde los servicios web desplegados por la Api de Sidbrint creada y modificada a partir del módulo service y módulos creados y agregados mediante el mismo modulo service.

## **Chart.js**

Chart JS es una herramienta open source bajo el licenciamiento del MIT, la cual permite crear y manipular gráficas de distintas tipologías. Herramienta basada en javascript.

Puede ser usada para crear gráficos elegantes y responsives sobre Canvas HTML5.

Permite mezclar diferentes tipos de gráficos y trazar datos en escalas fecha tiempo, logarítmica, o personalizada con facilidad. También soporta animaciones que pueden ser aplicadas cuando se cambian los datos o se actualizan colores.

ChartJS librería que me ha permitido desplegar gran parte de las visualizaciones graficas del TFG como son las gráficas en barras y el tipo donut, permitiendo de una manera práctica presentar los datos obtenidos por las llamas Ajax para su manejo como leyendas y data que se presenta en cada una de las gráficas.

## **Guzzle**

Guzzle es un cliente HTTP de PHP que facilita el envío de solicitudes HTTP y la integración sencilla con los servicios web.

Interfaz sencilla para crear cadenas de consulta, solicitudes POST, transmisión de grandes cargas, transmisión de grandes descargas, uso de cookies HTTP, carga de datos JSON, etc.

Puede enviar solicitudes síncronas y asíncronas utilizando la misma interfaz.

Utiliza interfaces PSR-7 para solicitudes, respuestas y flujos. Esto le permite utilizar otras bibliotecas compatibles con PSR-7 con Guzzle.

Extrae el transporte HTTP subyacente, lo que le permite escribir el entorno y el transporte de código agnóstico; es decir, no depende en gran medida de cURL, flujos de PHP, sockets o bucles de eventos no bloqueantes.

El sistema de middleware le permite aumentar y componer el comportamiento del cliente.

Guzzle en el caso de este TFG ha sido usada para ser parte de la capa de comunicación entre la aplicación web y los servicios presentados por la Api Sidbrint. Gracias a su característica de manejo de puertos y peticiones http, guzzle ha sido un poderoso aliado para el despliegue de las request enviadas y response obtenidas desde los servicios de la Api.

## **JavaScript**

JavaScript (abreviado comúnmente JS) es un lenguaje de programación interpretado, dialecto del estándar ECMAScript. Se define como orientado a objetos, basado en prototipos, imperativo, débilmente tipado y dinámico.

Se utiliza principalmente en su forma del lado del cliente (client-side), implementado como parte de un navegador web permitiendo mejoras en la interfaz de usuario y páginas web dinámicas aunque existe una forma de JavaScript del lado del servidor (Server-side JavaScript o SSJS). Su uso en aplicaciones externas a la web, por ejemplo en documentos PDF, aplicaciones de escritorio (mayoritariamente widgets) es también significativo.

Desde el 2012, todos los navegadores modernos soportan completamente ECMAScript 5.1, una versión de JavaScript. Los navegadores más antiguos soportan por lo menos ECMAScript 3. La sexta edición se liberó en julio del 2015.

JavaScript se diseñó con una sintaxis similar a C, aunque adopta nombres y convenciones del lenguaje de programación Java. Sin embargo, Java y JavaScript tienen semánticas y propósitos diferentes.

Todos los navegadores modernos interpretan el código JavaScript integrado en las páginas web. Para interactuar con una página web se provee al lenguaje JavaScript de una implementación del Document Object Model (DOM).

Tradicionalmente se venía utilizando en páginas web HTML para realizar operaciones y únicamente en el marco de la aplicación cliente, sin acceso a funciones del servidor. Actualmente es ampliamente utilizado para enviar y recibir información del servidor junto con ayuda de otras tecnologías como AJAX. JavaScript se interpreta en el agente de usuario al mismo tiempo que las sentencias van descargándose junto con el código HTML.

Desde el lanzamiento en junio de 1997 del estándar ECMAScript 1, han existido las versiones 2, 3 y 5, que es la más usada actualmente (la 4 se abandonó). En junio de 2015 se cerró y publicó la versión ECMAScript

## **Drupal**

### **Instalación de módulos:**

Drupal tiene la capacidad de obtener funcionalidad adicional mediante el uso de módulos. Los módulos pueden proporcionar muchas características adicionales diferentes, como pancartas, formularios, foros e informes. Son muy parecidos a los complementos en otros sistemas de administración de contenido como WordPress y Joomla.

Se pueden encontrar módulos que realicen tareas específicas buscando en [drupal.org](http://drupal.org) el tipo de módulo adecuado. La instalación se realiza mediante el panel de Drupal.

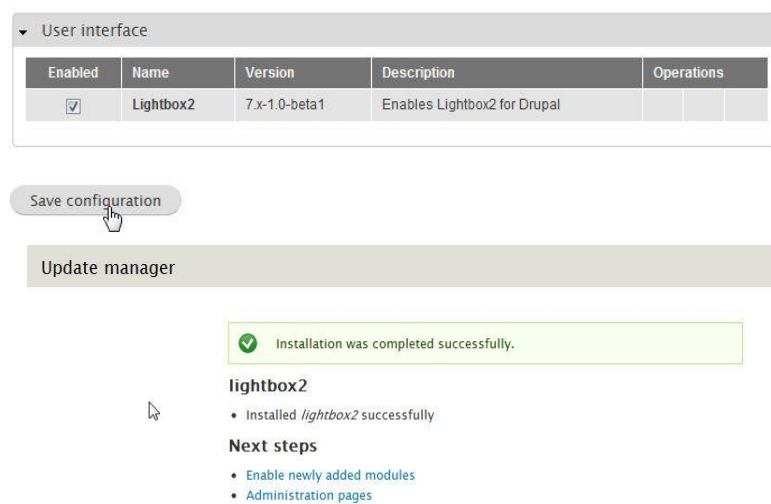
#### **1 Pasos:**

- Después de encontrar un módulo que desea instalar, descárguelo en el ordenador local.
- Inicie sesión en el panel de Drupal 7
- Ir a módulos
- Haga click en Instalar nuevo módulo
- Haga click en Elegir archivo y seleccione el módulo que descargó en su ordenador local
- 
- Haga click en instalar

## 2 Habilitar Modulo:

- Ahora, si desea habilitar el nuevo módulo, haga clic en el enlace Habilitar módulos recién agregados
- Desplace hacia abajo, busque el módulo deseado y coloque una casilla de verificación junto a él para habilitar el módulo.

## 3 Haga clic en Guardar configuración



- **Configuración:**

Ir a la sección → `admin/structure/services`

Agregar nuevo servicio.

Home » Administration » Structure

## Services

Services are collections of methods available to remote applications. They are defined in modules, and may be accessed in a number of ways through the REST API. **Use the "Add" button to add a new service.**

All enabled services and methods are shown. Click on any method to view information or test.

[+ Add](#) [+ Import](#)

Storage:  Enabled:  Search:

Sort by:  Order:

**These form fields let you filter and sort the list of services shown below.**

NAME	STORAGE	OPERATIONS
service1	Overridden	<a href="#">Edit Resources</a>
test_service	Normal	<a href="#">Edit Resources</a>

**This listing is for a Drupal website that has defined two services, named "service1" and "test\_service." Each service can be further defined by using its "Edit Resources" button.**

Todos los servicios creados son definidos como endpoints sobre los cuales se publicarán distintas funcionalidades/recursos la cuales son en realidad los servicios que se presentan mediante la publicación web.

### Add a new endpoint

Machine-readable name of the endpoint \*

The endpoint name can only consist of lowercase letters, underscores, and numbers.

Server \*

☒ - Select -  
☐ REST  
☐ XMLRPC

server that should be used to handle requests to this endpoint.

Path to endpoint \*

☐ Debug mode enabled  
Useful for developers. Do not enable on production environments

Authentication

☐ Session authentication

Choose which authentication schemes that should be used with your endpoint. If no authentication method is selected all requests will be done by an anonymous user.

Las distintas funcionalidades pueden ser generadas por código o creando vistas sobre las cuales se visualizarán resultados de consultas generales.

En nuestro caso las funcionalidades o recursos que se presentan han sido creados por código y agregando nuevos módulos dentro de la estructura de módulos de DRUPAL.



- Configuración de las vistas (*views*) necesarias para poder extraer toda la información que existe actualmente en la página web (brigadistas, encuadres militares y fuentes documentales). Estas vistas deberían extraerse en formato JSON.
- Las vistas son usadas como su nombre lo indica para dar la visualización de los distintos datos, apartados entre otros en drupal y distintos frameworks de desarrollo web.
- Preguntas claves para la definición del funcionamiento de una vista:
  - Le gusta la vista de página frontal predeterminada, pero encuentra que desea clasificarla de manera diferente.
  - Le gusta la taxonomía / vista de término predeterminada, pero encuentra que desea clasificarla de manera diferente; por ejemplo, alfabéticamente.
  - Usted usa / tracker, pero quiere restringirlo a publicaciones de cierto tipo.
  - Le gusta la idea del módulo 'artículo', pero no muestra los artículos como le gusta.
  - Desea una forma de mostrar un bloque con las 5 publicaciones más recientes de algún tipo en particular.
  - Quieres proporcionar 'mensajes de foro no leídos'.
  - Desea un archivo mensual similar a los archivos típicos de Movable Type / Wordpress que muestra un enlace al "Mes, YYYY (X)", donde X es el número de publicaciones ese mes y las muestra en un bloque. Los enlaces llevan a una simple lista de publicaciones para ese mes.

Las vistas pueden ser entendidas como la parte visual de la web, sin embargo, pueden tener otros usos para los desarrolladores como por ejemplo para unirlos con el módulo service, para preparar lugares donde servir los datos en diferentes estructuras como podrían ser en formato JSON; por cada servicio creado se correspondería una vista específica.

En esta parte se agregó por código los métodos que dan el servicio para las consultas necesarias y su presentación mediante los puntos de acceso. No es requerido usar vistas para poder dar acceso a los recursos de los servicios presentados. En nuestro caso se han realizado las funcionalidades sin usar vistas.

Instalación y configuración de un sistema de autenticación segura OAuth, para que las llamadas funcionen con un protocolo de seguridad y a través de una Auth Key y una Secret Key.

Agregar Módulo OAuth:

- Descargar OAUTH 1.0
- Load zip mediante herramienta para agregar módulos en drupal.
- Activar módulos para instalar.

	NOM	DESCRIPCIÓN
<input type="checkbox"/>	<b>OAuth</b> (oauth_common)	► Provides OAuth functionality
<input checked="" type="checkbox"/>	<b>OAuth Authentication</b> (services_oauth)	► Provides OAuth authentication for the services module
<input checked="" type="checkbox"/>	<b>OAuth Provider UI</b> (oauth_common_providerui)	► Provides a UI for when OAuth is acting as a provider.

### Configuración:

En OAUTH agregar un nuevo contexto, el cual será la referencia entre el user sobre el cual se genera el CONSUMER\_KEY y el ACCESS\_KEY los cuales permitirán la conexión hacia el servidor de rest api para la validación de credenciales. Usando estas dos credenciales permite tener una configuración de autenticación mediante dos puntos. O lo que es llamado (**2-legged OAuth**), para tener acceso a lo que se denomina (**3-legged OAuth**), se requiere tener como recurso otro servidor adicional y las claves de TOKEN y TOKEN\_SECRET

### Definición de nuevo módulo:

Se ha de crear una carpeta con el nombre del módulo que se agregara para ser usado por el endpoint.

```
custom_brigad_services
```

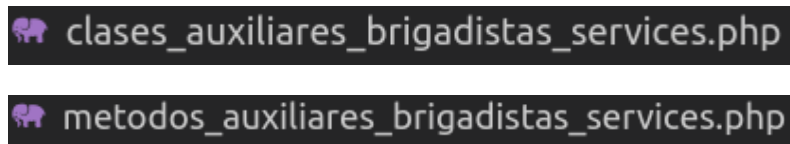
### Creación de ficheros primordiales para el uso del nuevo módulo:

Como nombre contendrán el mismo que se puso a la carpeta del nuevo módulo pero se tendrán tres tipos de extensiones (info, module, inc)

```
custom_brigad_services.info
custom_brigad_services.module
custom_brigad_services.inc
```

- **.info**  
El fichero info contendrá la descripción del módulo, path entre otros datos descriptivos del módulo.
- **.module**  
El fichero module contendrá la definición del módulo donde se incluirá la referencia al fichero .inc para la presentación de los recursos que se presentaran para publicar mediante los endpoint.
- **.inc**  
El fichero inc contendrá la definición de cada recurso, con los métodos y ficheros de clases incluidos para el manejo de cada petición a cada recurso.

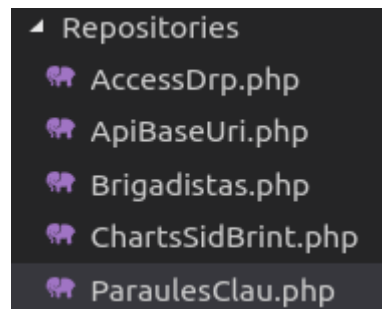
Para este caso se incluyeron algunos ficheros externos dentro del .inc que se muestran en la imagen a continuación:



```
clases_auxiliares_brigadistas_services.php
metodos_auxiliares_brigadistas_services.php
```

- **Desarrollo WEB.**

### Repositories



```

└─ Repositories
   ├── AccessDrp.php
   ├── ApiBaseUri.php
   ├── Brigadistas.php
   ├── ChartsSidBrint.php
   └── ParaulesClau.php

```

En repositories se mantiene la estructura de llamadas a los servicios y la validación de las estructuras de datos.

ApiBaseUri =>

Clase donde se define la uri base para la conexión respecto a las uris pertenecientes a los servicios presentados por el módulo service en Drupal

AccessDrp =>

Clase que donde se define la conexión respecto a los distintos métodos HTTP request y response por POST y GET.

Brigadistas =>

Clase donde se define el tratamiento de la petición y respuesta sobre servicios que permiten obtener información simple de brigadistas.

ChartsSidbrint =>

Clase donde se define el tratamiento para las peticiones y respuestas otorgadas por servicios que mantienen una estructura JSON que permite manejar la información para ser presentada mediante distintas graficas.

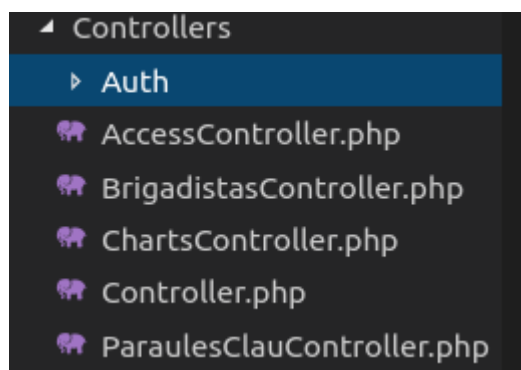
ParaulesClau =>

Clase para poder consultar la totalidad de palabras claves existentes dentro de la base de datos de Sidbrint.

Ejemplo:

En ApiBaseUri se encuentra la definición de la llama de autorización entre Laravell y Drupal en Sidbrint.

- **Controllors:**

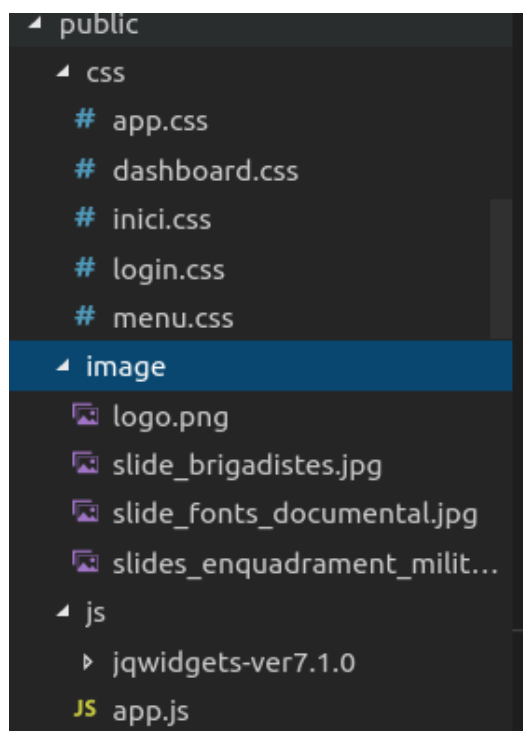


Los controladores proporcionan la manera de manejar la lógica de programación para poder administrar las peticiones realizadas a las distintas rutas dentro de la aplicación. En los controllers tenemos los distintos métodos que permiten derivar los request para ser derivados y a su vez modelar la respuesta dependiendo de la petición otorgada. En algunos casos puede ser directamente una vista o una respuesta con data modelada en estructuras JSON.

Por ejemplo:

ChartsController tiene el método `getIdeologiasPoliticas` el cual responde con una serie de estructuras JSON para ser modeladas por algunas de las librerías de JavaScript mencionadas en apartado superior donde se mencionan las tecnologías usadas para este proyecto.

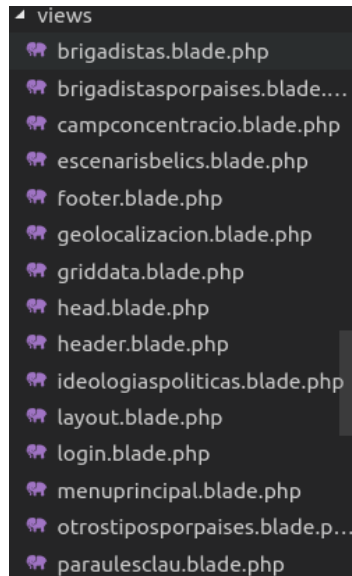
- **Public:**



Lugar donde se almacenan cosas de uso común, como son imágenes y ficheros de estilos usados en algunas de las vistas del proyecto. Así también se definen clases en las cuales se definen estructuras para modelar datos en las vistas.

- **Vistas**

Las vistas mantienen la visualización de la información de las estructuras JSON que recogemos de los servicios realizados en el módulo service de Drupal, en las distintas vistas sean usado las distintas tecnologías mencionadas, sobre las cuales se puede mencionar para interactuar con el DOM de la vista a JavaScript y JQuery.



- **Layout page:**

Es el layout de la página web. En este se tiene las regiones donde se muestran las distintas funcionalidades o páginas parciales.

Mantiene la estructura de la página principal de la web.

- **ROUTES:**

Las rutas definen como se trata una request o petición usando el patrón site/controller/funcionalidad

Que a su vez pueden ser llamadas por técnicas de desarrollo como AJAX donde se puede llamar a rutas funcionales que permitan obtener resultados como páginas, páginas parciales y resultados en formatos como lo son JSON entre otros.

Las rutas definen el tratamiento de las URLs siguiendo la estructura de end point o dominio de servicios, controlador y funcionalidad.

Rutas definidas a nivel web:

```
Sidbrint.edu/apirest/loginRoute::get('/login', function () {return view('login');
});
Route::get('/menuprincipal', function () {return view('menuprincipal');
})->name('home.index');
Route::get('/brigadistas', 'BrigadistasController@getAllBrigadistas');
```

```

Route::get('/paraulesclau', 'ParaulesClauController@getAllParaulesClau');
Route::post('loginAccess','AccessController@postResponseLogin')>name('loginAccess.post'
);
Route::get('/chartpagebrigadistas','ChartsController@getPageBrigadistasPorPaises');
Route::get('chartbrigadistasporpaises','ChartsController@getBrigadistasPorPaises');
Route::get('/geolocalizacion','BrigadistasController@getgeolocalizacion');
Route::get('geolocalizacionbrigadistasnacimiento','BrigadistasController@getBrigadistasGe
olocalizacionNacimiento');
Route::get('/chartpageideologias','ChartsController@getIdeologias');
Route::get('chartideologiasbrigadistas','ChartsController@getJsonDataToChartPieByParentI
d');
Route::get('/chartpagecampsconcentracion','ChartsController@getCampsConcentracion');
Route::get('chartCampsConcentracion','ChartsController@getJsonDataToChartPieByParentId
');
Route::get('/chartpageProfessions','ChartsController@getViewProfessions');
Route::get('/chartpageEscenarisBelics','ChartsController@getViewEscenarisBelics');
Route::get('chartsAll','ChartsController@getJsonDataToChartPieByParentId');
Route::get('searchdatabyids','BrigadistasController@getJsonDataToTableFind');
Route::get('/otrostipos','ChartsController@getOtrosTiposPorPaises');
Route::get('otrastipologias','ChartsController@getOtrtosTipos'); etc.

```

## Resultados

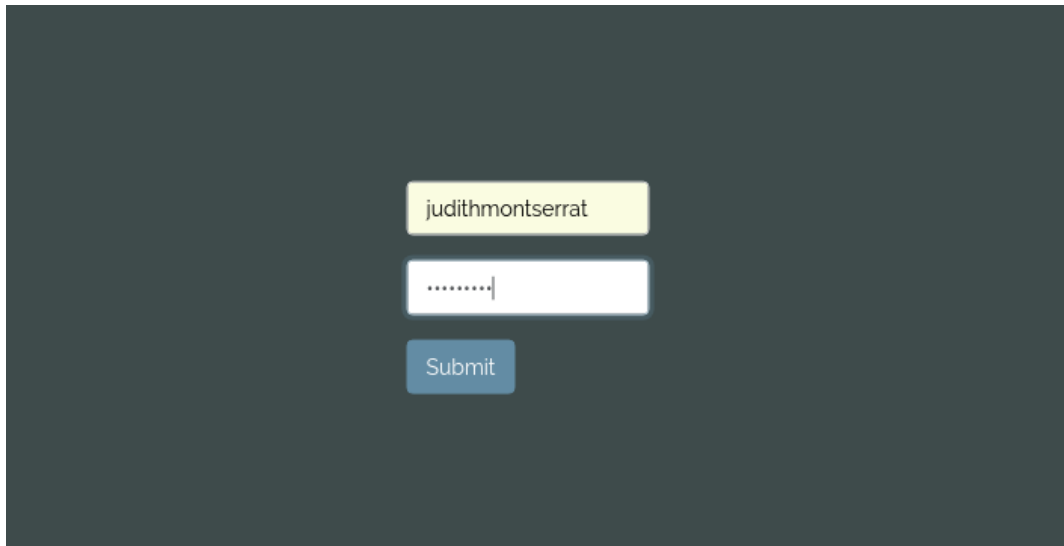
A continuación se muestran las distintas pantallas donde se hace uso de cada uno de los servicios publicados en la API.

Login user => Usando los usuarios de drupal para poder acceder.

Login Access:

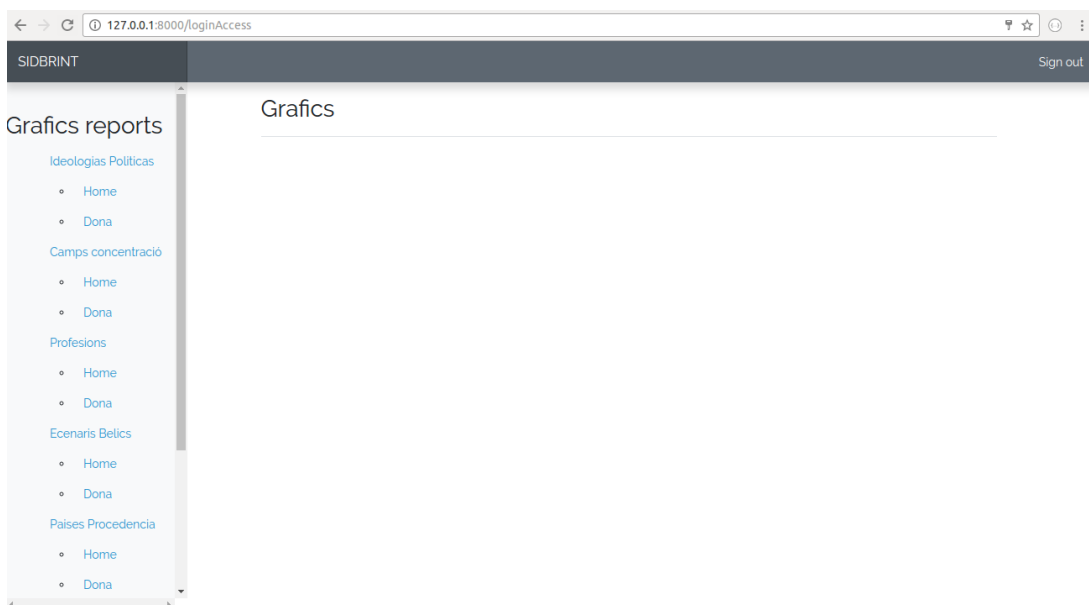


Mediante el api se publica el servicio de login donde se usarán las credenciales del usuario almacenado en drupal el actual gestor de contenidos de Sidbrint.

A screenshot of a login form on a dark grey background. It features a text input field containing the username 'judithmontserrat', a password input field with masked characters '.....', and a blue 'Submit' button below them.

Layout Page.

Página principal donde se alojarán los tags que usan las distintas rutas para invocar los distintos servicios publicados por la Api Sidbrint.





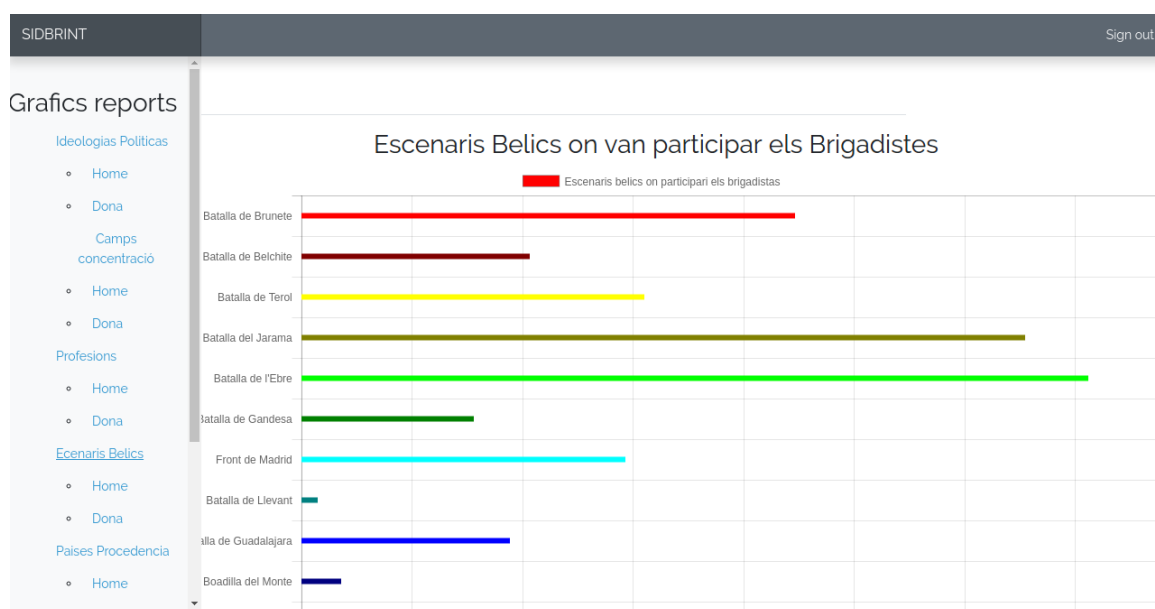
Usando servicios publicados en ApiRest Sidbrint.

Servicio: **numero\_Brigadistas\_PorGenero\_PalaraClave**

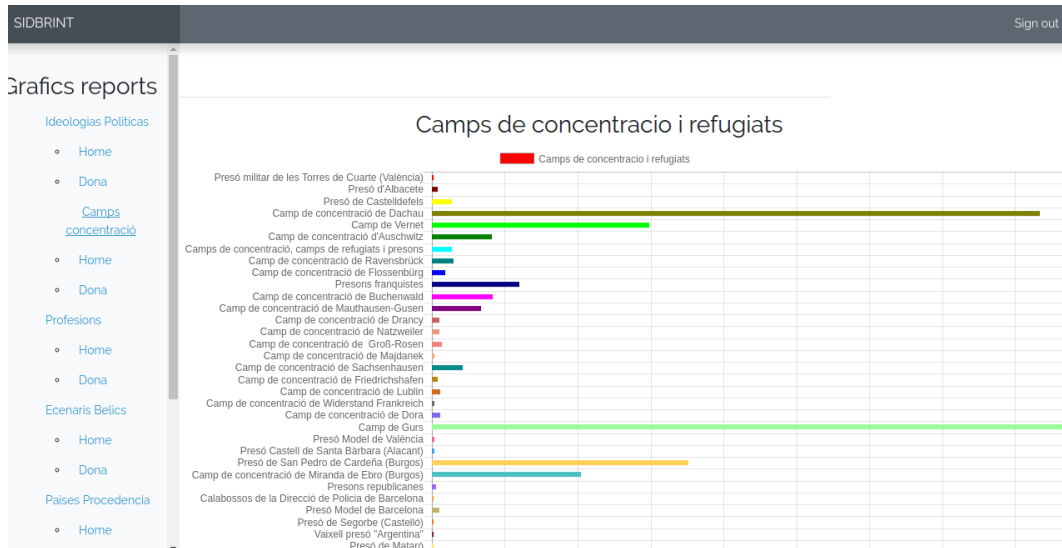
En la imagen inferior se grafica la información obtenida respecto a la cantidad total de brigadistas por ideología política que estuvieron en la guerra civil española. Con este valor se podría fundamentar que la guerra civil española pudo contener una índole política y que predominaron ciertas ideologías que caracterizaron la época.



En la imagen inferior se representa la cantidad de escenarios bélicos y cuantos brigadistas estuvieron asignados en cada escenario bélico, así respecto a los datos presentados se podría en cuales de los escenarios bélicos hubo una mayor participación por parte de los brigadistas extranjeros, así como también se podría plantear la hipótesis de que en los escenarios bélicos donde exista mayor participación de brigadistas fueron los que tenía mayor importancia ideológica.



Campos de concentración, en esta imagen se puede observar la cantidad de brigadistas que estuvieron en los distintos campos de concentración, con estos datos se puede interpretar cuales fueron los campos con mayor relevancia a la hora de contener a los distintos brigadistas por su importancia. Así como también si se modifica el servicio podría brindar una información más detallada agregando en el resultado las ideologías políticas y países de origen de los brigadistas contenidos en cada uno de los campos de concentración.



## Servicio: geolocalizacion\_brigadistas

La geolocalización por país de nacimiento que se muestra en la figura siguiente puede ser interpretada de manera directa, afirmando la cantidad de países que colaboraron en la guerra civil española y desde cuales vinieron más brigadistas. Con esta información se podría análogamente buscar las distintas ideologías políticas que existieron en dichos países para poder afirmar que por una razón ideología fue lo que motivo que gran cantidad de brigadistas provinieran de los países presentados.



En las imágenes en la parte inferior donde se puede observar la geolocalización por país de origen respecto a cada género, esto permite identificar directamente de que país provinieron la gran mayoría de mujeres y hombres que lucharon en la guerra civil española. Datos relevantes para los colaboradores que pudiesen requerir una estadística por genero de las distintas palabras clave referenciadas en el portal de Sidbrint.

Mujeres:

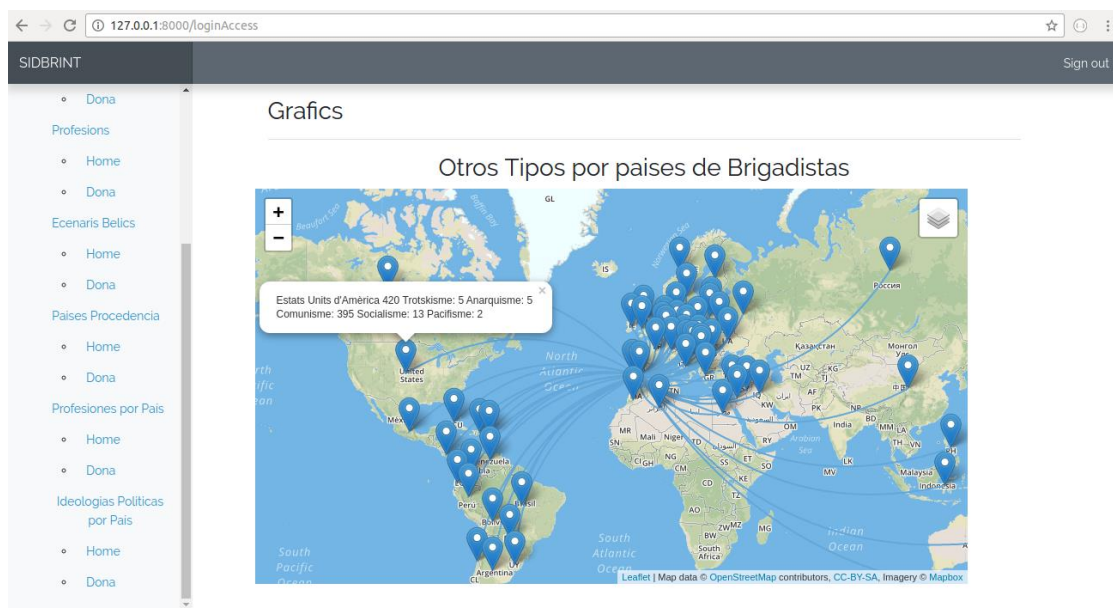


Hombres:



Servicio: numero\_BrigadistasGeolocalizados\_PaisNacimiento\_PalabraClave

En las imágenes que se presentan a continuación, se presenta como información por país de origen las ideologías políticas de cada país y la cantidad brigadistas que contuvo cada una, así como también las distintas profesiones que tuvieron los brigadistas en su país de origen al venir a participar en la guerra civil española.



## Conclusiones

En el proyecto se ha estipulado la creación de una API que ayude en la comunicación entre Sidbrint y sus colaboradores brindando la posibilidad de poder obtener información que pueda ser consumida y visualizada de maneras alternativas de acuerdo al uso que cada consumidor despliegue según la tecnología usada; todo esto me ha permitido obtener distintos conocimientos respecto al uso y despliegue de las API. Así como también las innumerables tecnologías que pueden ser usadas de acuerdo a las características que se mantiene en las actuales soluciones.

Los conocimientos adquiridos debido a cada tecnología me han enriquecido a nivel técnico y sobre los cuales me gustaría poder potenciar en un futuro.

El estar involucrado en todas las fases de desarrollo del proyecto también me ha hecho ver que los trabajos de análisis y documentación son tareas complejas que resultan altamente útiles para agilizar el desarrollo.

## Trabajo Futuro

Este proyecto se ha creado como un primer paso de una API la cual podría tener un largo recorrido. El haber creado una API permite que se puedan integrar otras aplicaciones a esta misma, como podrían ser aplicaciones nativas para dispositivos móviles.

Algunas de las mejoras que se podrían implementar en un futuro en la API serían:

### Seguridad

Modificación en Drupal para que toda la solución en Drupal y los servicios puedan ser llamados por el protocolo SSL para mayor seguridad.

### Añadir funcionalidades

Agregar nuevos servicios para poder realizar otras operaciones que puedan ser necesarias para los distintos casos que se puedan presentar entre consultas y mantenimientos futuros de los datos incluidos en Sidbrint.

### Modificación de parámetros

Incluir parámetros adicionales que pueda realizar una variación a las consultas actuales, por ejemplo un `parameters[País]` para identificar un único país.

## Referencias

Drupal Database Queris

<https://www.drupal.org/docs/7/api/database-api/dynamic-queries>

Laravel

<https://laravel.com/>

Bootstrap

<https://getbootstrap.com/>

Lealeft

<https://leafletjs.com/>

JQuery

<https://jquery.com/>

HTML5 Tutorial

[https://www.w3schools.com/html/html5\\_intro.asp](https://www.w3schools.com/html/html5_intro.asp)

CSS Tutorial

<https://www.w3schools.com/css/>

Chartsjs

<https://www.chartjs.org/>

Guzzle

<http://docs.guzzlephp.org/en/stable/overview.html>

JavaScript

<https://www.javascript.com/>